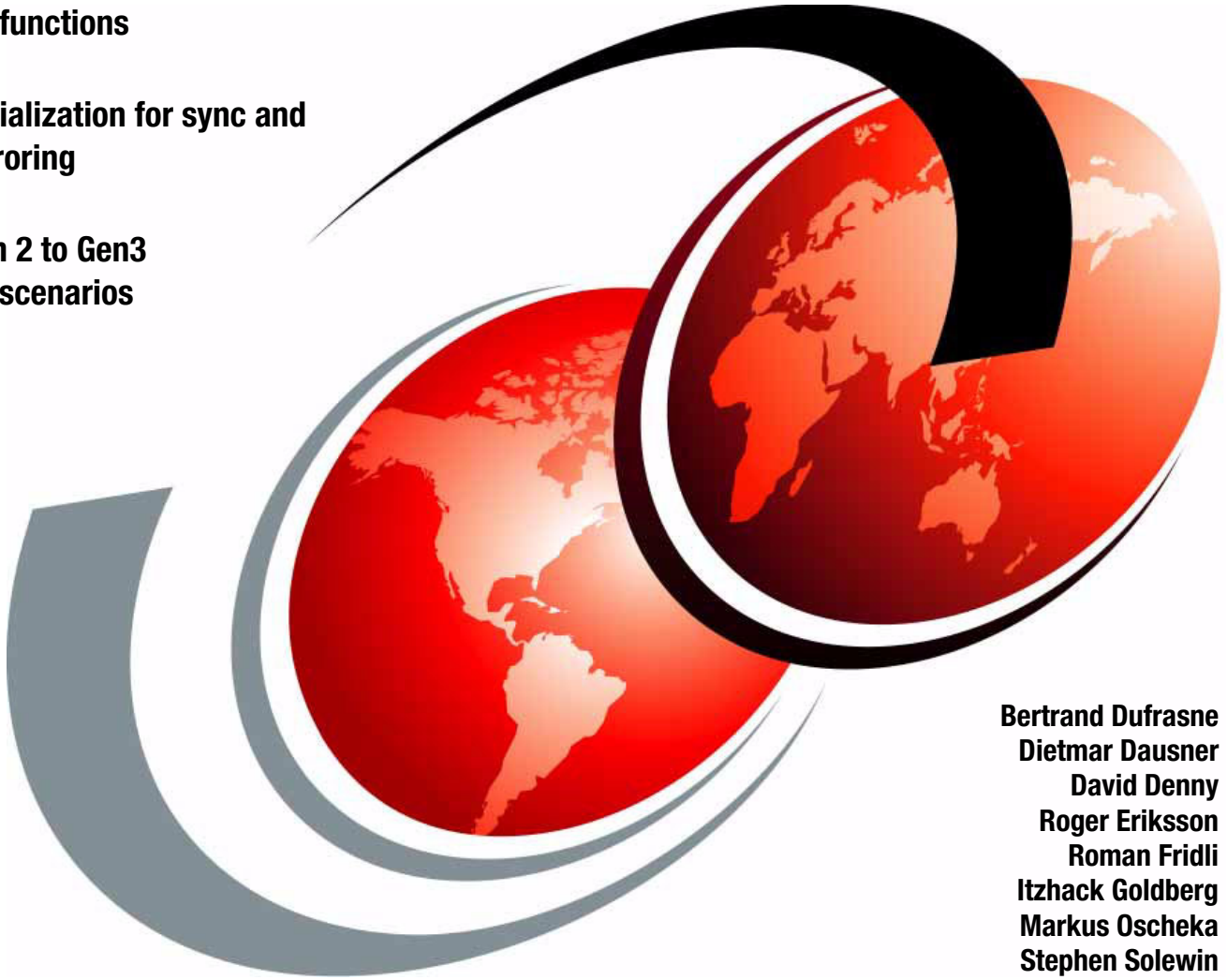
IBM

IBM XIV Storage System Copy Services and Migration

- Details of the copy services and migration functions
- Offline initialization for sync and async mirroring
- Generation 2 to Gen3 migration scenarios



Bertrand Dufrasne
Dietmar Dausner
David Denny
Roger Eriksson
Roman Fridli
Itzhack Goldberg
Markus Oscheka
Stephen Solewin

Redbooks

ibm.com/redbooks

International Technical Support Organization

IBM XIV Storage System Copy Services and Migration

February 2014

SG24-7759-04

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

Fifth Edition (February 2014)

This edition applies to Version 11.4 of the IBM XIV Storage System Software and Version 3.4 of the IBM XIV Storage System Hardware. Note however that some illustrations might still reflect older versions of the XIV GUI.

© Copyright International Business Machines Corporation 2014. All rights reserved.
Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices ix
Trademarks x

Preface xi
Authors xi
Now you can become a published author, too! xiii
Comments welcome xiii
Stay connected to IBM Redbooks xiii

Summary of changes xv
February 2014, Fifth Edition xv

Chapter 1. XIV Copy Services introduction 1

Chapter 2. Snapshots 3
2.1 Snapshots architecture 4
2.2 .Snapshot handling 11
2.2.1 Creating a snapshot 11
2.2.2 Viewing snapshot details 14
2.2.3 Deletion priority 15
2.2.4 Restore a snapshot 17
2.2.5 Overwriting snapshots 18
2.2.6 Unlocking a snapshot 19
2.2.7 Locking a snapshot 21
2.2.8 Deleting a snapshot 22
2.2.9 Automatic deletion of a snapshot 23
2.3 Snapshots consistency group 24
2.3.1 Creating a consistency group 24
2.3.2 Creating a snapshot using consistency groups 28
2.3.3 Managing a consistency group 29
2.3.4 Deleting a consistency group 32
2.4 Snapshot with remote mirror 33
2.5 MySQL database backup example 34
2.6 Snapshot example for a DB2 database 40
2.6.1 XIV Storage System and AIX OS environments 40
2.6.2 Preparing the database for recovery 41
2.6.3 Using XIV snapshots for database backup 42
2.6.4 Restoring the database from the XIV snapshot 42

Chapter 3. Volume copy 45
3.1 Volume copy architecture 46
3.2 Performing a volume copy 46
3.2.1 Monitoring the progress of a volume copy 48
3.3 Troubleshooting issues with volume copy 48
3.3.1 Using previously used volumes 48
3.4 Cloning boot volumes with XIV volume copy 49

© Copyright IBM Corp. 2014. All rights reserved.

Chapter 4. Remote mirroring 53
4.1 XIV Remote mirroring overview 54
4.1.1 XIV remote mirror terminology 54
4.1.2 XIV remote mirroring modes 56
4.2 Mirroring schemes 58
4.2.1 Peer designations and roles 59
4.2.2 Operational procedures 60
4.2.3 Mirroring status 62
4.3 XIV remote mirroring usage 65
4.3.1 Using snapshots 67
4.4 XIV remote mirroring actions 69
4.4.1 Defining the XIV mirroring target 69
4.4.2 Setting the maximum initialization and synchronization rates 74
4.4.3 Connecting XIV mirroring ports 74
4.4.4 Defining the XIV mirror coupling and peers: Volume 75
4.4.5 Activating an XIV mirror coupling 80
4.4.6 Adding volume mirror coupling to consistency group mirror coupling 81
4.4.7 Normal operation: Volume mirror coupling and CG mirror coupling 82
4.4.8 Deactivating XIV mirror coupling: Change recording 83
4.4.9 Changing role of slave volume or CG 84
4.4.10 Changing role of master volume or CG 84
4.4.11 Mirror reactivation and resynchronization: Normal direction 86
4.4.12 Synchronous mirror deletion and using offline initialization for resynchronization 86
4.4.13 Reactivation, resynchronization, and reverse direction 87
4.4.14 Switching roles of mirrored volumes or CGs 87
4.4.15 Adding a mirrored volume to a mirrored consistency group 87
4.4.16 Removing a mirrored volume from a mirrored consistency group 89
4.4.17 Deleting mirror coupling definitions 90
4.5 Best practice usage scenarios 91
4.5.1 Failure at primary site: Switch production to secondary 91
4.5.2 Complete destruction of XIV 1 92
4.5.3 Using an extra copy for DR tests 93
4.5.4 Creating application-consistent data at both local and remote sites 93
4.5.5 Migration through mirroring 94
4.5.6 Migration using Hyper-Scale Mobility 94
4.5.7 Adding data corruption protection to disaster recovery protection 94
4.5.8 Communication failure between mirrored XIV systems 95
4.5.9 Temporary deactivation and reactivation 95
4.5.10 Connectivity type change 96
4.5.11 Mirror type conversion 96
4.5.12 Volume resizing across asynchronous XIV mirror pairs 97
4.6 Planning 97
4.7 Advantages of XIV mirroring 98
4.8 Mirroring events 98
4.9 Mirroring statistics for asynchronous mirroring 98
4.10 Boundaries 99
4.11 Using the GUI or XCLI for remote mirroring actions 99
4.11.1 Initial setup 99
4.11.2 Remote mirror target configuration 104
4.11.3 XCLI examples 111
4.12 Configuring remote mirroring 112

Chapter 5. Synchronous Remote Mirroring 113
5.1 Synchronous mirroring considerations 114
5.2 Setting up mirroring 115
5.2.1 Using the GUI for volume mirroring setup 115
5.2.2 Using the GUI for volume mirror activation 118
5.2.3 Using XCLI for volume mirroring setup 120
5.2.4 Using XCLI for volume mirror activation 120
5.3 Setting up mirroring for a consistency group 121
5.3.1 Considerations regarding a consistency group 121
5.3.2 Using the GUI for CG mirroring setup 122
5.4 Mirrored snapshots (ad-hoc sync jobs) 126
5.4.1 Using the GUI for creating a mirrored snapshot 126
5.5 Mirror activation, deactivation, and deletion 128
5.6 Role reversal tasks (switch or change role) 129
5.6.1 Switch roles 129
5.6.2 Change role 131
5.7 Link failure and last consistent snapshot 132
5.7.1 Last-consistent snapshot (LCS) 133
5.7.2 Last-consistent snapshot timestamp 133
5.7.3 External last consistent snapshot (ELCS) 134
5.8 Disaster recovery cases 135
5.8.1 Disaster recovery scenario with synchronous mirroring 135

Chapter 6. Asynchronous remote mirroring 149
6.1 Asynchronous mirroring configuration 150
6.1.1 Volume mirroring setup and activation 150
6.1.2 Consistency group configuration 158
6.1.3 Coupling activation, deactivation, and deletion 161
6.2 Role reversal 165
6.2.1 Switching roles 165
6.2.2 Change role 166
6.3 Resynchronization after link failure 168
6.3.1 Last-replicated snapshot 169
6.4 Disaster recovery 169
6.5 Mirroring process 170
6.5.1 Initialization process 170
6.5.2 Ongoing mirroring operation 172
6.5.3 Mirroring consistency groups 172
6.5.4 Mirrored snapshots 172
6.5.5 Mirroring special snapshots 174
6.6 Detailed asynchronous mirroring process 176
6.7 Asynchronous mirror step-by-step illustration 178
6.7.1 Mirror initialization 178
6.7.2 Remote backup scenario 179
6.7.3 DR testing scenario 180
6.8 Pool space depletion 184

Chapter 7. Open systems considerations for Copy Services 187
7.1 AIX specifics 188
7.1.1 AIX and snapshots 188
7.1.2 AIX and Remote Mirroring 191
7.2 Copy Services using VERITAS Volume Manager 193
7.2.1 Snapshots with VERITAS Volume Manager 193

7.2.2 Remote Mirroring with VERITAS Volume Manager 195
7.3 HP-UX and Copy Services 196
7.3.1 HP-UX and XIV snapshot 196
7.3.2 HP-UX with XIV Remote Mirror 197
7.4 Windows and Copy Services 198
7.4.1 Windows Volume Shadow Copy Service with XIV Snapshot 199
7.5 VMware virtual infrastructure and Copy Services 207
7.5.1 Virtual machine considerations regarding Copy Services 207
7.5.2 VMware ESX server and snapshots 208
7.5.3 ESX and Remote Mirroring 214

Chapter 8. IBM i considerations for Copy Services 217
8.1 IBM i functions and XIV as external storage 218
8.1.1 IBM i structure 218
8.1.2 Single-level storage 218
8.1.3 Auxiliary storage pools (ASPs) 219
8.2 Boot from SAN and cloning 219
8.3 Our implementation 220
8.4 Snapshots with IBM i 222
8.4.1 Solution benefits 223
8.4.2 Disk capacity for the snapshots 223
8.4.3 Power-down IBM i method 224
8.4.4 Quiescing IBM i and using snapshot consistency groups 227
8.4.5 Automation of the solution with snapshots 231
8.5 Synchronous Remote Mirroring with IBM i 232
8.5.1 Solution benefits 233
8.5.2 Planning the bandwidth for Remote Mirroring links 233
8.5.3 Setting up synchronous Remote Mirroring for IBM i 233
8.5.4 Scenario for planned outages 235
8.5.5 Scenario for unplanned outages 237
8.6 Asynchronous Remote Mirroring with IBM i 241
8.6.1 Benefits of asynchronous Remote Mirroring 241
8.6.2 Setting up asynchronous Remote Mirroring for IBM i 242
8.6.3 Scenario for planned outages and disasters 243

Chapter 9. Data migration 245
9.1 Overview 246
9.2 Handling I/O requests 248
9.3 XIV and source storage connectivity 249
9.3.1 Multipathing with data migrations 249
9.3.2 Understanding initialization rate 252
9.4 Data migration steps 253
9.4.1 Initial connection and pre-implementation activities 254
9.4.2 Perform pre-migration tasks for each host being migrated 261
9.4.3 Define and test data migration volume 262
9.4.4 Activate a data migration on XIV 266
9.4.5 Define the host on XIV and bring host online 266
9.4.6 Complete the data migration on XIV 268
9.4.7 Post migration activities 269
9.5 Command-line interface 269
9.5.1 Using XCLI scripts or batch files 273
9.5.2 Sample scripts 274
9.6 Manually creating the migration volume 274

9.7 Changing and monitoring the progress of a migration 276
9.7.1 Changing the synchronization rate 276
9.7.2 Monitoring migration speed 278
9.7.3 Monitoring the impact of migration on host latency 278
9.7.4 Monitoring migration through the XIV event log 279
9.7.5 Monitoring migration speed through the fabric 280
9.7.6 Monitoring migration speed through the source storage system 280
9.7.7 Predicting run time using actual throughput 280
9.8 Thick-to-thin migration 281
9.9 Resizing the XIV volume after migration 282
9.10 Migrating XIV Generation 2 to XIV Gen3 285
9.10.1 Generation 2 to Gen3 migration using XDMU 285
9.10.2 Generation 2 to Gen3 migration using replication 285
9.10.3 Generation 2 to Gen3 migration in multi-site environments 286
9.10.4 Server- based migrations 290
9.11 Troubleshooting 296
9.11.1 Target connectivity fails 296
9.11.2 Remote volume LUN is unavailable 297
9.11.3 Local volume is not formatted 298
9.11.4 Host server cannot access the XIV migration volume 298
9.11.5 Remote volume cannot be read 299
9.11.6 LUN is out of range 299
9.12 Backing out of a data migration 300
9.12.1 Back out before migration is defined on the XIV 300
9.12.2 Back out after a data migration has been defined but not activated 300
9.12.3 Back out after a data migration has been activated but is not complete 300
9.12.4 Back out after a data migration has reached the synchronized state 300
9.13 Migration checklist 301
9.14 Device-specific considerations 304
9.14.1 EMC CLARiiON 305
9.14.2 EMC Symmetrix and DMX 306
9.14.3 HDS TagmaStore USP 307
9.14.4 HP EVA 307
9.14.5 IBM DS3000, DS4000, DS5000 308
9.14.6 IBM ESS 800 310
9.14.7 IBM DS6000 and DS8000 312
9.14.8 IBM Storwize V7000 and SAN Volume Controller 313
9.14.9 N series and iSCSI setup 313
9.15 Host-specific considerations 314
9.15.1 VMware ESX 315
9.16 Sample migration 322

Chapter 10. Using Tivoli Storage Productivity Center for Replication 331
10.1 IBM Tivoli Productivity Center family 332
10.2 What Tivoli Productivity Center for Replication provides 332
10.3 Supported operating system platforms 333
10.4 Copy Services terminology 333
10.4.1 Copy sets 335
10.4.2 Sessions 336
10.5 Session states 337
10.6 System and connectivity overview 338
10.7 Monitoring 339
10.8 Web interface 340

10.8.1 Connecting to Tivoli Productivity Center for Replication GUI 340
10.8.2 Health Overview panel 340
10.8.3 Sessions panel 342
10.8.4 Storage Subsystems panel 342
10.9 Defining and adding XIV storage 343
10.10 XIV snapshots 345
10.10.1 Defining a session for XIV snapshots 345
10.10.2 Defining and adding copy sets to a session 348
10.10.3 Activating Snapshot session 350
10.10.4 Additional snapshot actions inside a session 353
10.11 XIV synchronous mirroring (Metro Mirror) 354
10.11.1 Defining a session for Metro Mirror 354
10.11.2 Defining, adding copy sets to a Metro Mirror session 358
10.11.3 Activating a Metro Mirror session 362
10.11.4 Suspending the Metro Mirror (XIV Synchronous Mirror) session 364
10.12 XIV asynchronous mirrors (Global Mirror) 368
10.12.1 Defining a session for asynchronous mirroring (Global Mirror) 368
10.12.2 Defining and adding copy sets to a Global Mirror session 370
10.12.3 Activating the Global Mirror session 370
10.12.4 Suspending the Global Mirror session 370
10.13 Using Tivoli Productivity Center for Replication to add XIV Volume Protection . . . 370

Related publications 373
IBM Redbooks publications 373
Other publications 373
Online resources 374
Help from IBM 374

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

© Copyright IBM Corp. 2014. All rights reserved.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®
AS/400®
BladeCenter®
DB2®
DS4000®
DS6000™
DS8000®
FlashCopy®
i5/OS™
IBM®
iSeries®
POWER®
Power Systems™
Redbooks®
Redpaper™
Redbooks (logo) ®
Storwize®
System i®
System p®
System Storage®
System x®
Tivoli®
WebSphere®
XIV®
z/OS®

The following terms are trademarks of other companies:

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

LTO, the LTO Logo and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Preface

This IBM® Redbooks® publication provides a practical understanding of the IBM XIV® Storage System copy and migration functions. The XIV Storage System has a rich set of copy functions suited for various data protection scenarios, which enables clients to enhance their business continuance, data migration, and online backup solutions. These functions allow point-in-time copies, known as snapshots and full volume copies, and also include remote copy capabilities in either synchronous or asynchronous mode. These functions are included in the XIV software and all their features are available at no additional charge.

The various copy functions are reviewed in separate chapters, which include detailed information about usage, and also practical illustrations.

The book also illustrates the use of IBM Tivoli® Storage Productivity Center for Replication to manage XIV Copy Services.

This book is intended for anyone who needs a detailed and practical understanding of the XIV copy functions.

Note: GUI and XCLI illustrations included in this book were created with an early release of the version 11.4 code, as available at the time of writing. There might be minor differences with the XIV version 11.4 code that is publicly released.

Authors

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, San Jose Center.

Bertrand Dufrasne is an IBM Certified Consulting I/T Specialist and Project Leader for IBM System Storage® disk products at the International Technical Support Organization, San Jose Center. He has worked in a variety of I/T areas at IBM. Bertrand has authored many IBM Redbooks publications and has also developed and taught technical workshops. Before joining the ITSO, he was an Application Architect in IBM Global Services. He holds a Master of Electrical Engineering degree.

Dietmar Dausner is a certified XIV Product Field Engineer in Germany for the EMEA region and joined IBM as a manufacturing test engineer for storage devices. Later, he became a customer application engineer for hard disk drives, supporting large OEM accounts in Europe. In 2007, he joined the European Storage Competence Center as a Product Field Engineer (PFE). Since 2008, he has supported the XIV Storage System. Dietmar holds a degree in Electrical Engineering.

David Denny is a Solutions Architect with XIV in the IBM Systems and Technology Group. David has over 20 years of experience in the IT field, ranging from systems administration to enterprise storage architect. David is the lead corporate resource for data migrations with XIV. Prior to joining IBM, David was a Lead Architect of the Enterprise SAN for the DoD Disaster Recovery Program at the Pentagon following the events of 9/11. He holds a Bachelor of Arts degree as well a Bachelor of Science degree in Computer Science from Lynchburg College.

Roger Eriksson is an STG Lab Services consultant, based in Stockholm, Sweden, who works for the European Storage Competence Center in Mainz, Germany. He is a Senior

© Copyright IBM Corp. 2014. All rights reserved.

Accredited IBM Product Service Professional. Roger has over 20 years of experience working on IBM servers and storage, including Enterprise and Midrange disk, NAS, SAN, IBM System x®, IBM System p®, and IBM BladeCenter®. He has consulted and performed proof of concepts and education, mainly with the XIV product line, since December 2008. He has worked with both clients and various IBM teams worldwide. He holds a Technical College Graduation in Mechanical Engineering.

Roman Fridli is a certified IBM XIV Product Field Engineer based in Switzerland. He joined IBM in 1998 as a Customer Engineer for IBM Power Systems™ and Intel Servers, including point-of-sales. Since 2012, he has worked for the XIV PFE EMEA-Team based in Mainz, Germany. He holds a degree in Electrical Engineering and multiple certifications in the storage solution and networking area.

Itzhack Goldberg is an IBM Technical Advisor in the EMEA region for the XIV Storage System, based in Haifa, Israel. Itzhack worked at the IBM Austin lab from 1989 to 1997, for the development of the IBM AIX® Logical Volume Manager and File System. He has won an award for the design and development of a data recovery suite for AIX. Following that assignment, Itzhack worked on the code-load design and development of the IBM DS6000™. He holds a degree in Computer Science.

Markus Oscheka is an IT Specialist for Proof of Concepts and Benchmarks with the Disk Solution Europe team in Mainz, Germany. His areas of expertise include the setup and demonstration of IBM System Storage solutions in various environments, such as IBM AIX, Linux, Microsoft Windows, VMware ESX, and Solaris. He has worked at IBM for 12 years, performing many proof of concepts and performance benchmarks for disk storage products. Markus has contributed and acted as co-project leader for DS8000 and XIV Redbooks. He holds a degree in Electrical Engineering from the Technical University in Darmstadt.

Stephen Solewin is an XIV Corporate Solutions Architect for IBM in Tucson, Arizona. He has 16 years of experience working on IBM storage, including Enterprise and Midrange Disk, LTO drives and libraries, SAN, storage virtualization, and storage software. Steve has worked on the XIV product line since March of 2008. He holds a BS degree in Electrical Engineering from the University of Arizona, where he graduated with honors.

Thanks to the authors of the previous edition: Desire Brival, Mark Kremkus, Thomas Peralto, Aubrey Applewhaite, David Denny, Jawed Iqbal, Christina Lara, Jana Jamsek, Lisa Martinez, Rosemary McCutchen, Hank Sautter, Stephen Solewin, Ron Verbeek, Roland Wolf, Eugene Tsypin, Roger Eriksson, Carlo Saba, Kip Wagner, Nils Nause, Alexander Warmuth, Axel Westphal, Wilhelm Gardt, Ralf Wohlfarth.

Special thanks to Rami Elron, Tim Dawson for their help and advice on many of the topics covered in this book.

Thanks to the following people for their contributions to this project:
Iddo Jacobi, Diane Benjuya, Moriel Lechtman, Brian Sherman, Juan Yanes.

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author - all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks
- Send your comments in an email to:

  redbooks@us.ibm.com
- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/pages/IBM-Redbooks/178023492563?ref=ts
- Follow us on twitter:

  http://twitter.com/ibmredbooks
- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806
- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm
- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

Summary of changes

This section describes the technical changes made in this edition of the book and in previous editions. This edition might also include minor corrections and editorial changes that are not identified.

Summary of Changes
for SG24-7759-04
for IBM XIV Storage System Copy Services and Migration
as created or updated on February 20, 2014.

February 2014, Fifth Edition

This revision reflects the addition, deletion, or modification of new and changed information described below.

New Information

New capabilities related to copy functions are introduced with the XIV Software v11.4:

- Protection from automatic deletion for snapshots in thin provisioned pools, in Chapter 2, “Snapshots” on page 3
- Offline initialization for synchronous mirrors, in Chapter 5, “Synchronous Remote Mirroring” on page 113.
- External last consistent snapshot in 5.7.3, “External last consistent snapshot (ELCS)” on page 134.
- The “Migrating XIV Generation 2 to XIV Gen3” on page 285 section is new.

For a description of the IBM Hyper-Scale Consistency function, see the IBM Redpaper™ publication, *IBM Hyper-Scale in XIV Storage,* REDP-5053.

Changed information

This edition reflects various updates and corrections. Substantial changes were made to Chapter 4, “Remote mirroring” on page 53 and Chapter 9, “Data migration” on page 245.

The SAN Volume Controller and IBM Storwize® V7000 content is moved in *XIV Gen3 with IBM System Storage SAN Volume Controller and Storwize V7000*, REDP-5063.

© Copyright IBM Corp. 2014. All rights reserved.

1

XIV Copy Services introduction

The XIV Storage System provides a rich set of copy services functions suited for various data protection scenarios, and enables clients to enhance their business continuance, data migration, and online backup solutions.

Clients receive exceptionally low total cost of ownership because snapshots, synchronous mirroring, asynchronous mirroring, and data migration are all included in the system software licensing. An intuitive and powerful graphical user interface (GUI) greatly simplifies management of the storage system, enabling rapid learning of management tasks, and efficient use of storage administrator time.

Release 11.4 of IBM XIV Storage System now includes these features:

- Protection from automatic deletion for snapshots in thin provisioned pools, as detailed in Chapter 2, “Snapshots” on page 3.
- Offline initialization for synchronous mirrors, as detailed in Chapter 5, “Synchronous Remote Mirroring” on page 113.

© Copyright IBM Corp. 2014. All rights reserved.

2

Snapshots

A *snapshot* is a point-in-time copy of a volume's data. The XIV snapshot uses several innovative technologies to ensure minimal impact on system performance. Snapshots are space-efficient, make use of pointers, and only contain partitions with data that has changed from the original volume. If a partition has not changed, both the snapshot and the volume point to the same partition. They efficiently share cache for common data, effectively working as a larger cache than would be the case with full data copies.

A *volume copy* is an exact copy of an existing volume, and is described in detail in Chapter 3., "Volume copy" on page 45.

© Copyright IBM Corp. 2014. All rights reserved.

2.1 Snapshots architecture

Before the discussion of snapshots, a short review of XIV's architecture is provided. For more information see *IBM XIV Storage System Architecture and Implementation*, SG24-7659.

The XIV system consists of several modules. A module is a Intel based server, running XIV code, with 12 disk drives, one or more processor cores and memory that acts as a distributed cache. All the modules are connected to each other through redundant internal switches. Certain modules (4 - 9 in a full 15-module XIV) contain Fibre Channel and Ethernet host bus adapters that enable the XIV to connect to the network and the host servers (Figure 2-1).

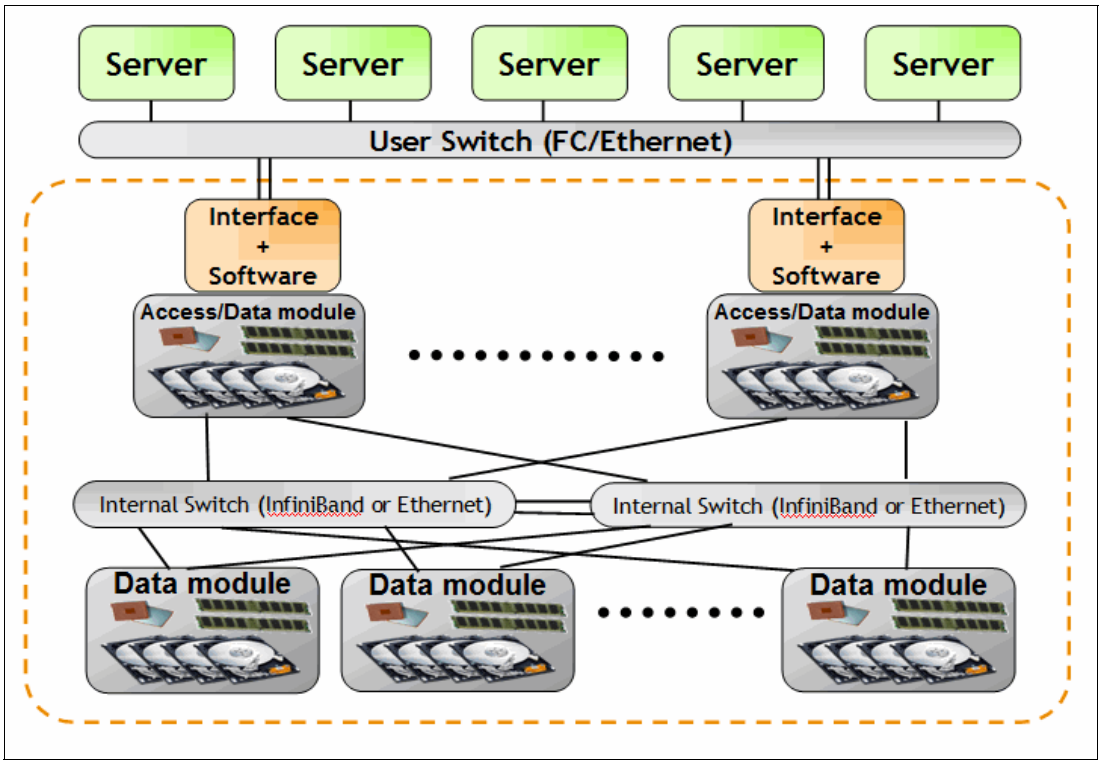


Figure 2-1 XIV architecture: Modules and disk drives

When a logical volume or LUN is created on an XIV system, the volume is divided into pieces that are 1 MB in size, called *partitions*. Each partition is duplicated for data protection and the two copies are stored on disks of different modules. All partitions of a volume are pseudo-randomly distributed across the modules and disk drives, as shown in Figure 2-2.

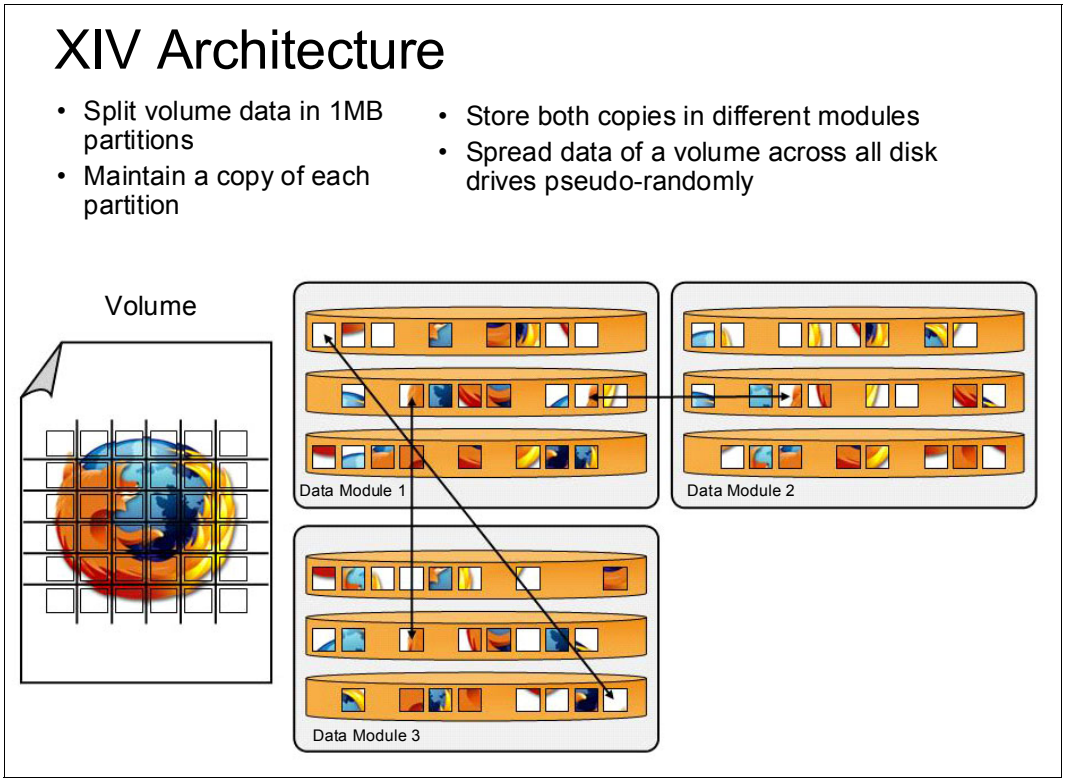

XIV Architecture
• Split volume data in 1MB partitions
• Maintain a copy of each partition
• Store both copies in different modules
• Spread data of a volume across all disk drives pseudo-randomly
Volume
Data Module 1
Data Module 2
Data Module 3

Figure 2-2 XIV architecture: Distribution of data

A logical volume is represented by pointers to partitions that make up the volume. If a snapshot is taken of a volume, the pointers are just copied to form the snapshot volume, as shown in Figure 2-3. No space is consumed for the snapshot volume at the time of snapshot creation.

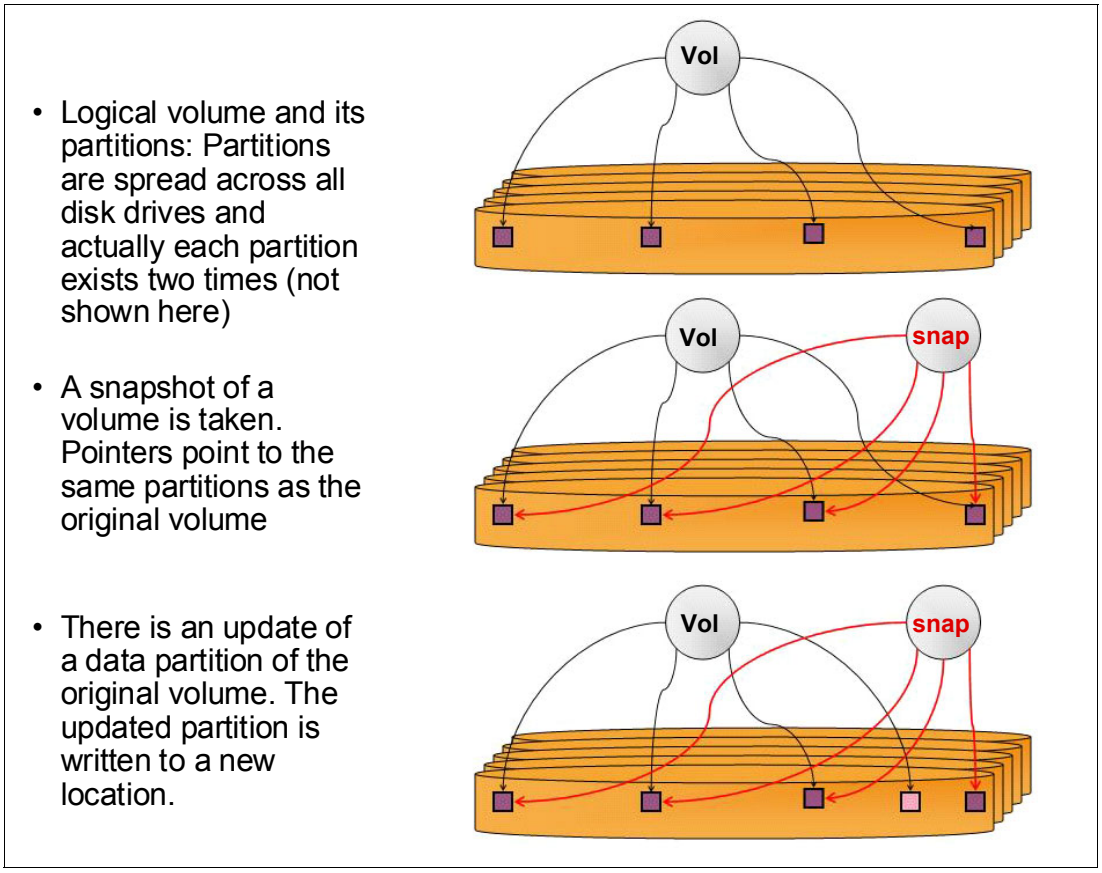

• Logical volume and its partitions: Partitions are spread across all disk drives and actually each partition exists two times (not shown here)
• A snapshot of a volume is taken. Pointers point to the same partitions as the original volume
• There is an update of a data partition of the original volume. The updated partition is written to a new location.
Vol
Vol
snap
Vol
snap

Figure 2-3 XIV architecture: Snapshots

When an update is done to the original data, the update is stored in a new partition and a pointer of the original volume now points to the new partition, but the snapshot volume still points to the original partition. This method is called *redirect-on-write*. Now we use up more space for the original volume and its snapshot, and it has the size of a partition (1 MB).

An important fact to remember is that a volume is more than just the information that resides on the physical disks. There is also metadata, the information about where that information resides. For XIV, metadata also indicates whether a specific partition has valid host information written on it. Metadata management is the key to rapid snapshot performance. A snapshot points to the partitions of its master volume for all unchanged partitions. When the data is modified, a new partition is allocated for the modified data, which means the XIV Storage System manages a set of pointers based on the volume and the snapshot. Those pointers are modified when changes are made to the user data. Managing pointers to data enables XIV to nearly instantly create snapshots, as opposed to physically copying the data into a new partition. See Figure 2-4 on page 7.

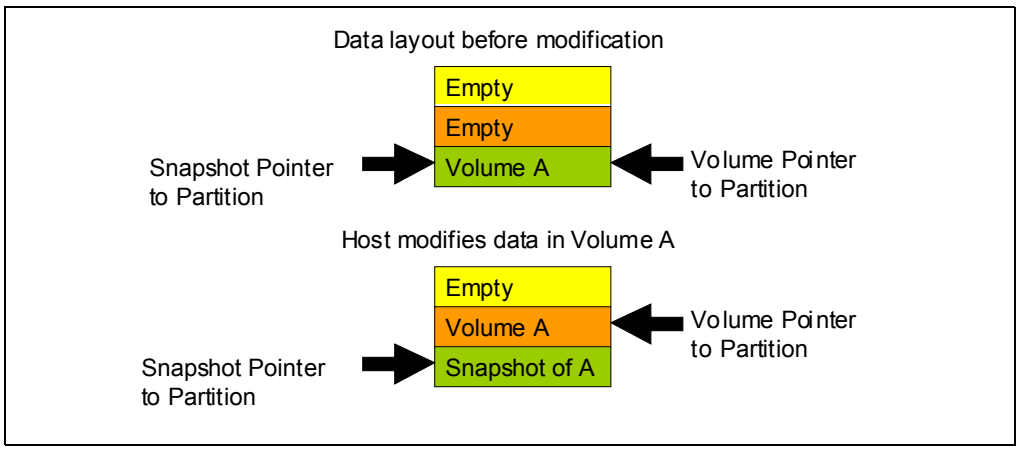


Figure 2-4 Example of a redirect-on-write operation

The actual metadata overhead for a snapshot is small. When the snapshot is created, the system does not require new pointers because the volume and snapshot are exactly the same, which means that the time to create the snapshot is independent of the size or number of snapshots present in the system. As data is modified, new metadata is created to track the changes to the data.

Note: The XIV system minimizes the impact to the host for write operations by performing a redirect-on-write operation. As the host writes data to a volume with a snapshot relationship, the incoming information is placed into a newly allocated partition. Then the pointer to the data for the master volume is modified to point at the new partition. The snapshot volume continues to point at the original data partition.

Because the XIV Storage System tracks the snapshot changes on a partition basis, data must only be coalesced when a transfer is less than the size of a partition. For example, a host writes 4 KB of data to a volume with a snapshot relationship. Although the 4 KB is written to a new partition, for the partition to be complete, the remaining data must be copied from the original partition to the newly allocated partition.

The alternative to redirect-on-write is the *copy on write* function. Most other systems do not move the location of the volume data. Instead, when the disk subsystem receives a change, it copies the volume's data to a new location for the point-in-time copy. When the copy is complete, the disk system commits the newly modified data. Therefore, each individual modification takes longer to complete because the entire block must be copied before the change can be made.

Storage pools and consistency groups

A *storage pool* is a logical entity that represents storage capacity. Volumes are created in a storage pool and snapshots of a volume are within the same storage pool. Figure 2-5 shows an example of a storage pool.

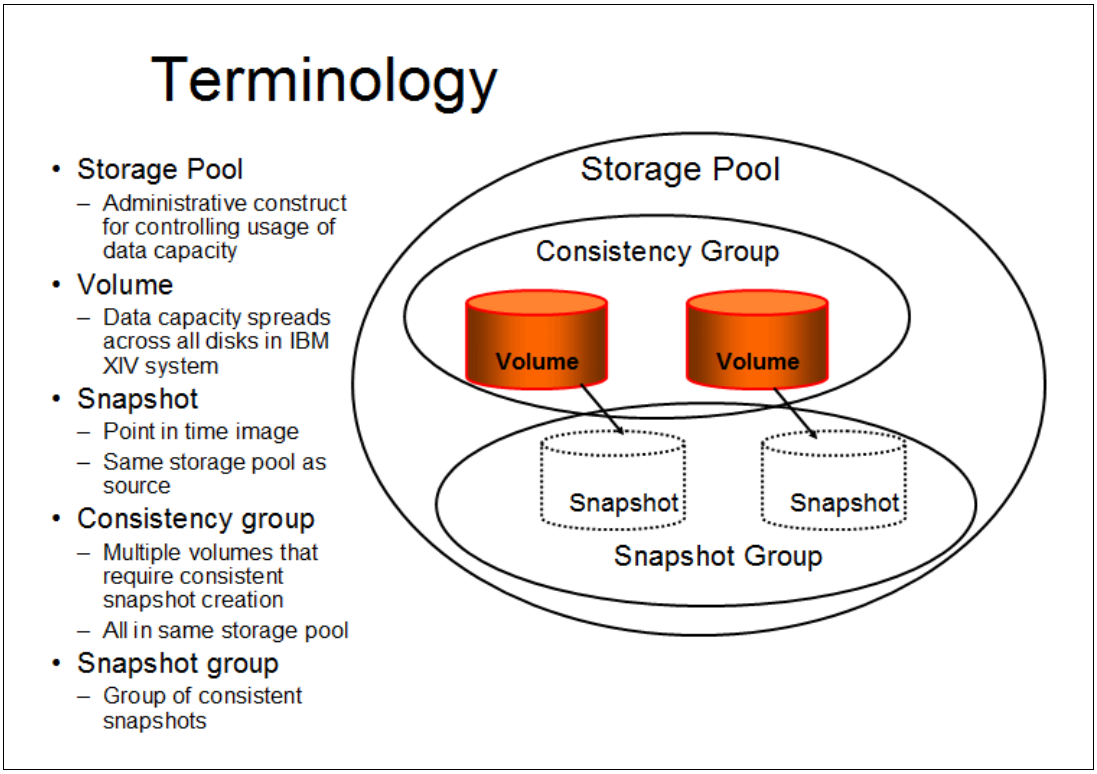


Figure 2-5 XIV terminology

Because snapshots require capacity as the source and the snapshot volume differ over time, space for snapshots must be set aside when defining a storage pool (Figure 2-6). A minimum of 34 GB of snapshot space should be allocated. The GUI defaults snapshot space to 10% of the total pool space. A value of 20% of the pool space is recommended for applications with higher write activity. A storage pool can be resized as needed if there is enough free capacity in the XIV Storage System.

Figure 2-6 Creating a storage pool with capacity for snapshots

An application can utilize many volumes on the XIV Storage System. For example, a database application can span several volumes for application data and transaction logs. In this case, the snapshot for the volumes must occur at the same moment in time so that the data and logs are consistent. The *consistency group* allows the user to perform the snapshot on all the volumes assigned to the group at the same moment, thereby enforcing data consistency.

Automatic snapshot deletion

The XIV Storage System implements an automatic snapshot deletion mechanism (see Figure 2-7 on page 10) to protect itself from overutilizing the snapshot space. Snapshot space overutilization can occur as a volume has new data written to it and also when new snapshots are created.

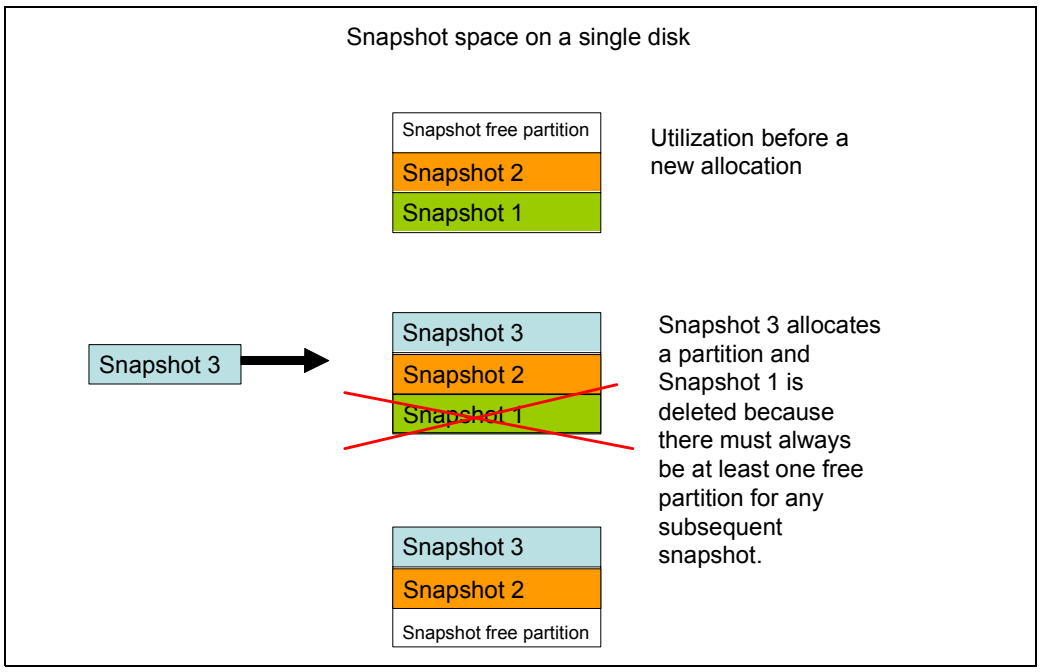

Snapshot space on a single disk
Snapshot free partition
Snapshot 2
Snapshot 1
Utilization before a new allocation
Snapshot 3
Snapshot 3
Snapshot 2
Snapshot 1
Snapshot 3 allocates a partition and Snapshot 1 is deleted because there must always be at least one free partition for any subsequent snapshot.
Snapshot 3
Snapshot 2
Snapshot free partition

Figure 2-7 Diagram of automatic snapshot deletion

Each snapshot has a deletion priority property that is set by the user. There are five deletion priorities; priority 4 is first and priority 1 is last. There is also a deletion priority of 0, which is allowed only in thin provisioned pools. Priority 0 is used to protect snapshots that must be immune to automatic deletion.The system uses this priority to determine which snapshot to delete first. When the system needs to delete a snapshot to make room for a new snapshot, it starts deleting the oldest snapshot with deletion priority 4 (first). The system will delete all the snapshots with priority first, before starting to delete snapshots with priority 3. All snapshots with deletion priority 2 are next, and finally, snapshots with a deletion priority of 1 (last). Snapshots with a deletion priority of 0 are not subject to automatic deletion. This is illustrated in 2.2.3, “Deletion priority” on page 15.

Important: Be sure to monitor snapshot space usage in the pool when setting deletion priority to 0. When pool hard-space is completely used, all writes to every volume in the pool will stop.

Manual deletion of snapshots is further explained in 2.2.8, “Deleting a snapshot” on page 22.

The XIV storage system provides alerts based upon percentage of the snapshot space being used. Figure 2-8 on page 11 shows a possible setting for Snapshots Usage that is different from the factory defaults. As you receive higher level alerts concerning snapshot usage. the snapshot space of the affected pool can be expanded to accommodate more snapshots.

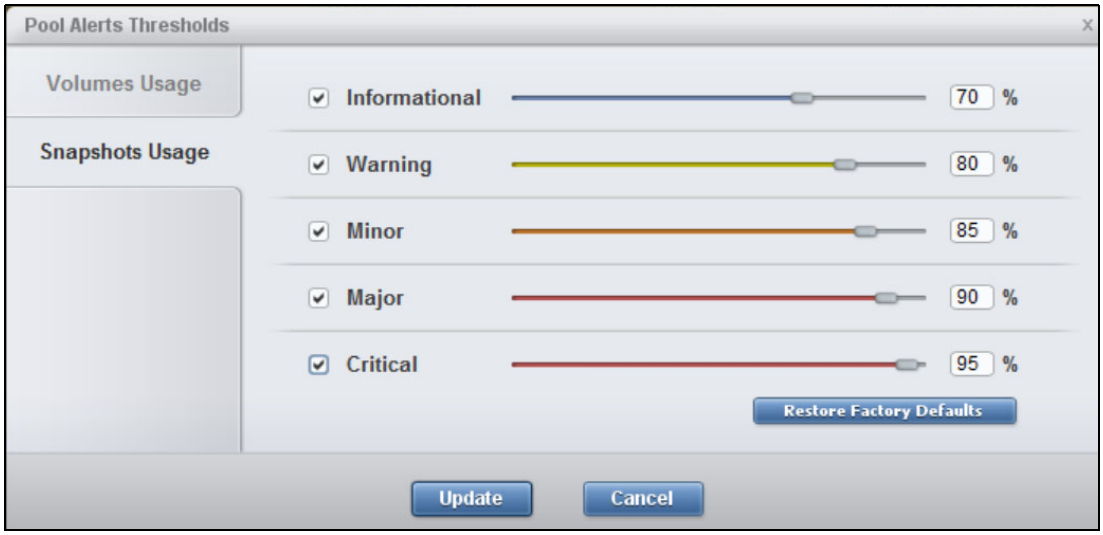


Figure 2-8 Snapshots Usage Pool Alert

XIV Asynchronous Mirroring leverages snapshot technology. See Chapter 6, “Asynchronous remote mirroring” on page 149 for details about XIV Asynchronous Mirroring.

2.2 .Snapshot handling

The creation and management of snapshots with the XIV Storage System is straightforward and easy to perform. This section guides you through the lifecycle of a snapshot, providing examples of how to interact with the snapshots using the GUI. This section also discusses duplicate snapshots and the automatic deletion of snapshots.

2.2.1 Creating a snapshot

Snapshot creation is a simple task to accomplish. Using the **Volumes and Snapshots** view, right-click the volume and select **Create Snapshot**. Figure 2-9 depicts how to make a snapshot of the `at12677_v3 volume`.

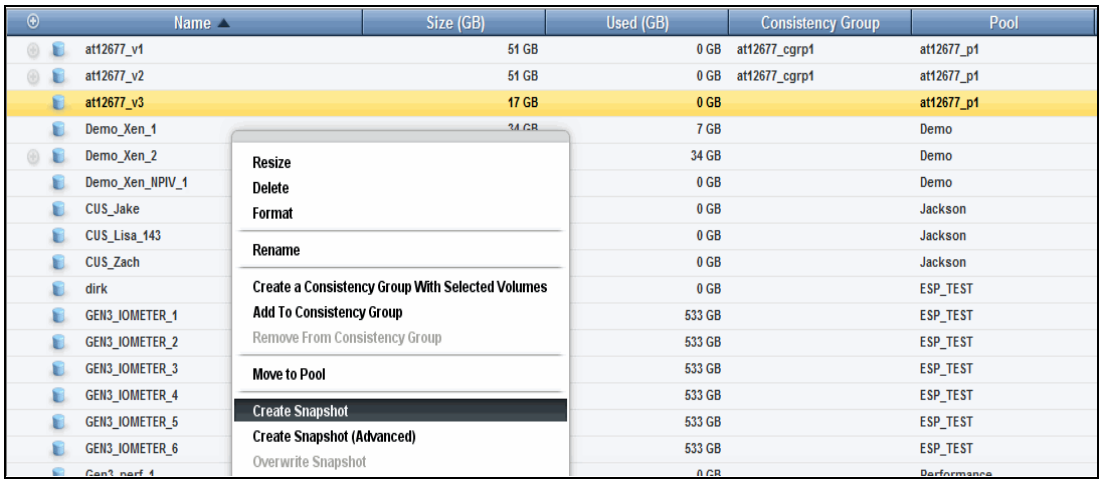


Figure 2-9 Creating a snapshot

The new snapshot is displayed in Figure 2-10. The XIV Storage System uses a specific naming convention. The first part is the name of the volume followed by the word *snapshot* and then a number or count of snapshots for the volume. The snapshot is the same size as the master volume. However, it does not display how much space has been used by the snapshot.

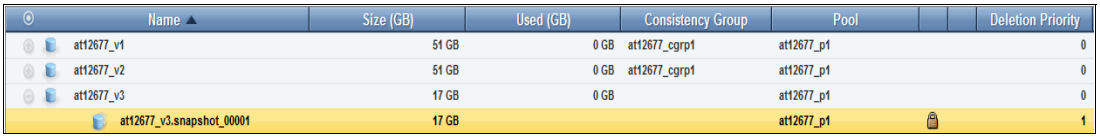


Figure 2-10 View of a new snapshot

From the view shown in Figure 2-10, other details are evident:

- First is the locked property of the snapshot. By default, a snapshot is locked, which means that it is read-only at the time of creation.
- Second, the modified property is displayed to the right of the locked property. In this example, the snapshot has not been modified.

You might want to create a duplicate snapshot—for example, if you want to keep this snapshot as is and still be able to modify a copy of it.

The duplicate has the same creation date as the first snapshot, and it also has a similar creation process. From the Volumes and Snapshots view, right-click the snapshot to duplicate. Select **Duplicate** from the menu to create a new duplicate snapshot. Figure 2-11 provides an example of duplicating the snapshot `at12677_v3.snapshot_00002`.

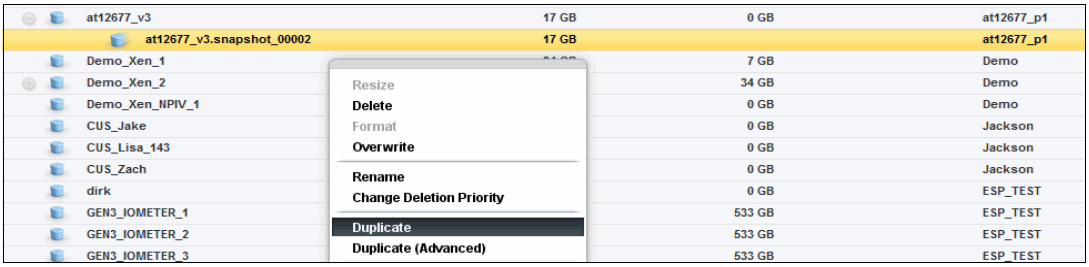


Figure 2-11 Creating a duplicate snapshot

After you select **Duplicate** from the menu, the duplicate snapshot is displayed directly under the original snapshot.

The Duplicate (Advanced) option allows you to change the name of the duplicated snapshot as seen in Figure 2-12 on page 13. The **Deletion Priority** displays the current deletion priority setting. This setting cannot be changed here. See 2.2.3, “Deletion priority” on page 15 for more information.

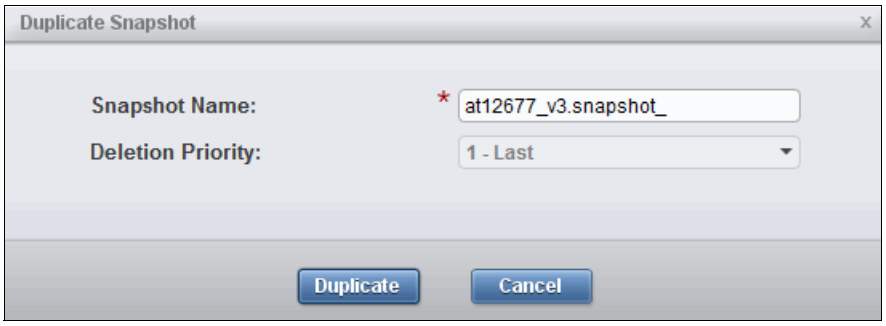


Figure 2-12 Duplicate (Advanced) option

Note: The creation date of the duplicate snapshot in Figure 2-13 is the same creation date as the original snapshot. The duplicate snapshot points to the master volume, not the original snapshot.

Figure 2-13 View of the new duplicate snapshot

Example 2-1 shows creation of a snapshot and a duplicate snapshot with the Extended Command Line Interface (XCLI).

The remaining examples in this section use the XIV session XCLI. You may also use the XCLI command. In this case, however, specify the configuration file or the IP address of the XIV that you are working with and also the user ID and password. Use the XCLI command to automate tasks with batch jobs. For simplicity, the examples use the XIV session XCLI.

Example 2-1 Creating a snapshot and a duplicate with the XCLI session

```
snapshot_create vol=ITSO_Volume
snapshot_duplicate snapshot=ITSO_Volume.snapshot_00001
```

After the snapshot is created, it must be mapped to a host to access the data. This action is performed in the same way as mapping a normal volume.

Important: A snapshot is an exact replica of the original volume. Certain hosts do not properly handle having two volumes with the same exact metadata describing them. In these cases, you must map the snapshot to a different host to prevent failures.

Creation of a snapshot is only done in the volume's storage pool. A snapshot cannot be created in a storage pool other than the one that owns the volume. If a volume is moved to another storage pool, the snapshots are moved with the volume to the new storage pool (provided that there is enough space).

2.2.2 Viewing snapshot details

After creating the snapshots, you might want to view the details of the snapshot for creation date, deletion priority, and whether the volume has been modified. Using the GUI, select **Snapshot Tree** from the Volumes menu, as shown in Figure 2-14.

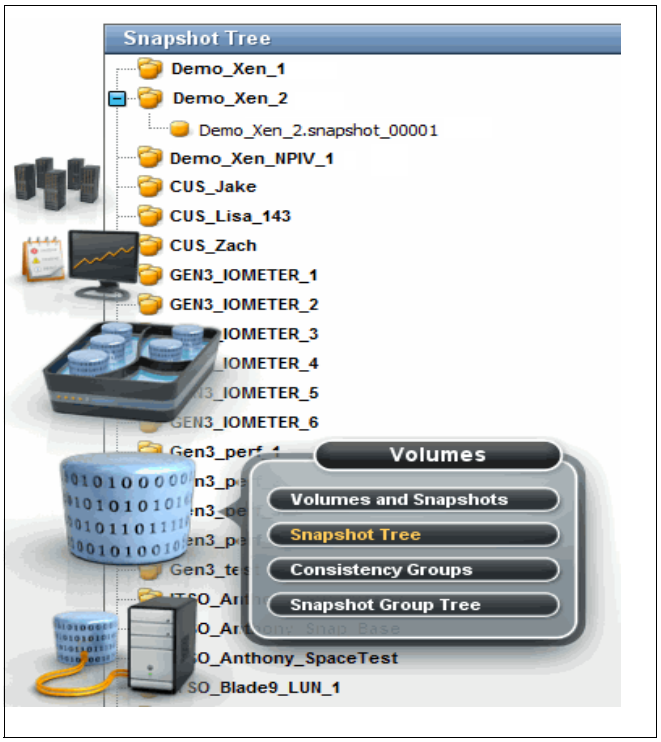


Figure 2-14 Selecting the Snapshot Tree view

The GUI displays all the volumes in a list.

Scroll to the snapshot of interest and select the snapshot by clicking its name. Details about the snapshot are displayed in the upper right panel. Looking at the volume `at12677_v3`, it contains a snapshot `00001` and a duplicate snapshot `00002`. The snapshot and the duplicate snapshot have the same creation date of `2011-09-02 12:07:49`, as shown in Figure 2-15. In addition, the snapshot is locked, has not been modified, and has a deletion priority of 1 (which is the highest priority, so it will be deleted last).

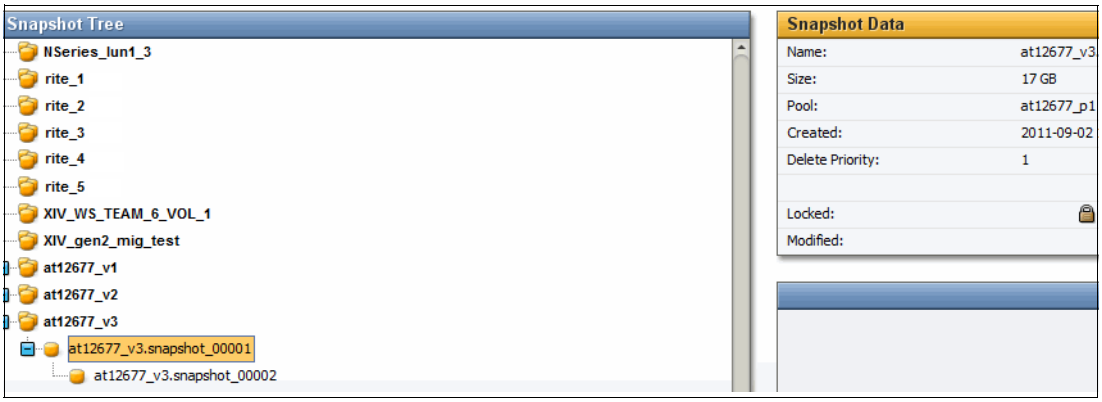


Figure 2-15 Viewing the snapshot details

Along with these properties, the tree view shows a hierarchal structure of the snapshots. This structure provides details about restoration and overwriting snapshots. Any snapshot can be overwritten by any parent snapshot, and any child snapshot can restore a parent snapshot or a volume in the tree structure.

In Figure 2-15 on page 14, the duplicate snapshot is a child of the original snapshot, or phrased another way, the original snapshot is the parent of the duplicate snapshot. This structure does not refer to the way the XIV Storage System manages the pointers with the snapshots, but is intended to provide an organizational flow for snapshots.

Example 2-2 shows the snapshot data output in the XCLI session. Because of space limitations, only a small portion of the data is displayed from the output.

Example 2-2 Viewing the snapshots with XCLI session

```
snapshot_list vol=ITSO_Volume

Name                          Size (GB)   Master Name   Consistency Group   Pool
ITSO_Volume.snapshot_00001    17          ITSO_Volume                       itso
ITSO_Volume.snapshot_00002    17          ITSO_Volume                       itso
```

2.2.3 Deletion priority

Deletion priority enables the user to rank the importance of the snapshots within a pool. For the current example, the duplicate snapshot `at12677_v3.snapshot_00002` is not as important as the original snapshot `at12677_v3.snapshot_00001`. Therefore, the deletion priority is reduced.

If the snapshot space is full, the duplicate snapshot is deleted first even though the original snapshot is older.

To modify the deletion priority, right-click the snapshot in the Volumes and snapshots view and select **Change Deletion Priority**, as shown in Figure 2-16.

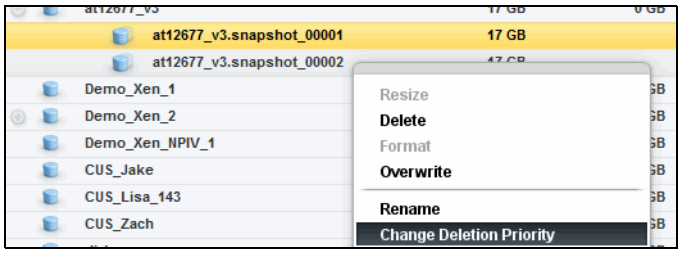


Figure 2-16 Changing the deletion priority

Then, select a deletion priority from the dialog window and click **OK** to accept the change.

Figure 2-17 shows the five options that are available for setting the deletion priority. The lowest priority setting is 4, which causes the snapshot to be deleted first. The highest priority setting is 1 for snapshots that are subject to automatic deletion, and these snapshots are deleted last. Deletion priority 0 is only allowed for thin provisioned pools, and is used to protect snapshots from automatic deletion. All snapshots have a default deletion priority of 1, if not specified on creation.

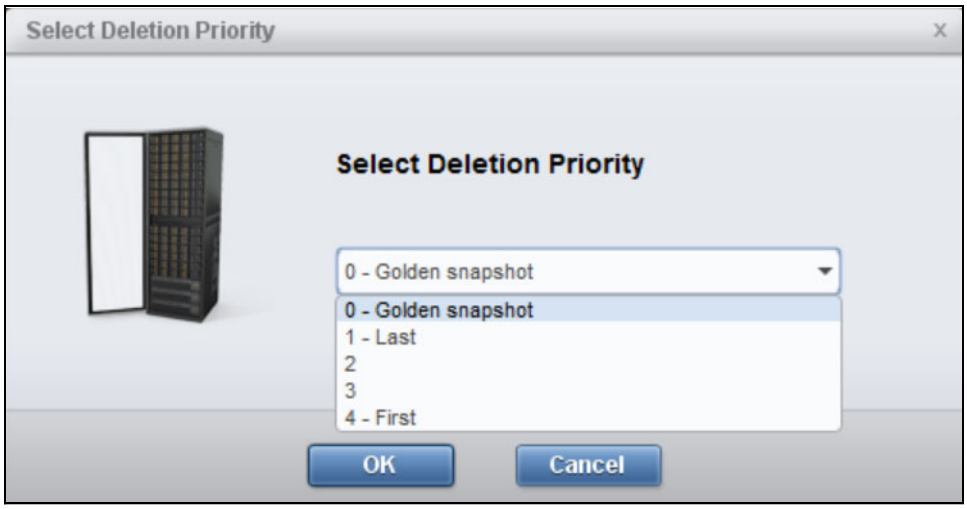


Figure 2-17 Changing snapshot deletion priority

Figure 2-18 displays confirmation that the duplicate snapshot has had its deletion priority lowered to 4. As shown in the upper right panel, the delete priority is reporting a 4 for snapshot `at12677_v3.snapshot_00001`.

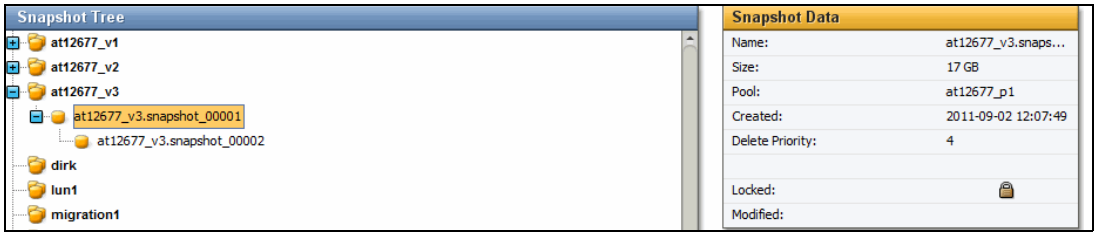


Figure 2-18 Confirming the modification to the deletion priority

To change the deletion priority for the XCLI session, specify the snapshot and new deletion priority, as illustrated in Example 2-3.

Example 2-3 Changing the deletion priority for a snapshot

```
snapshot_change_priority snapshot=ITSO_Volume.snapshot_00002 delete_priority=4
```

The GUI also lets you specify the deletion priority when you create the snapshot. Instead of selecting **Create Snapshot**, select **Create Snapshot (Advanced)**, as shown in Figure 2-19.

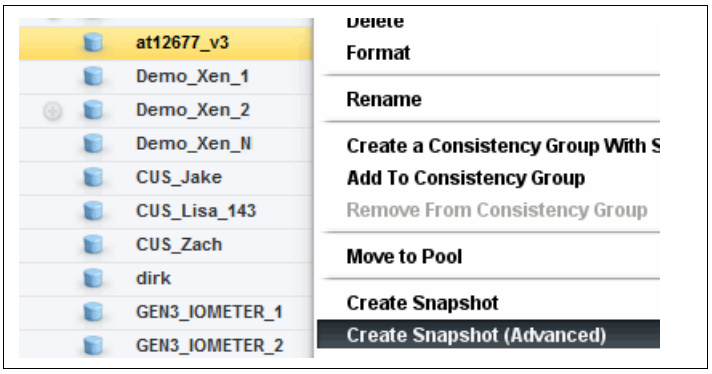


Figure 2-19 Create Snapshot (Advanced)

A panel is displayed that allows you to set the Snapshot Name and the Deletion Priority.

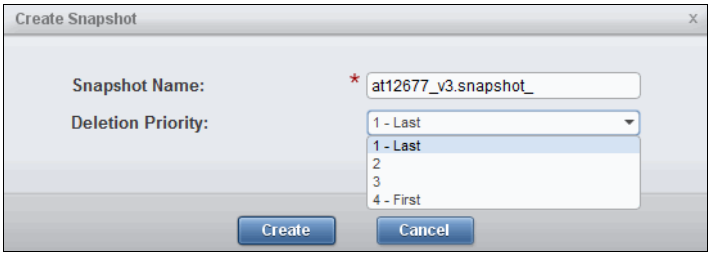


Figure 2-20 Advanced snapshot options

2.2.4 Restore a snapshot

The XIV Storage System provides the ability to restore the data from a snapshot back to the master volume, which can be helpful for operations where data was modified incorrectly and you want to restore the data. From the **Volumes and Snapshots** view, right-click the volume and select **Restore**. This action opens a dialog box where you can select which snapshot is to be used to restore the volume. Click **OK** to perform the restoration.

Figure 2-21 illustrates selecting the Restore action on the at12677_v3 volume.

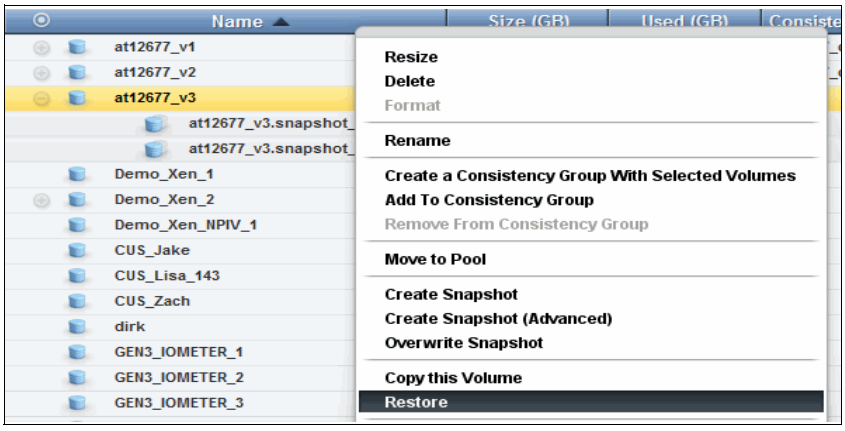


Figure 2-21 Snapshot volume restore

After you perform the restore action, you return to the **Volumes and Snapshots** panel. The process is instantaneous, and none of the properties (creation date, deletion priority, modified properties, or locked properties) of the snapshot or the volume have changed.

Specifically, the process modifies the pointers to the master volume so that they are equivalent to the snapshot pointer. This change only occurs for partitions that have been modified. On modification, the XIV Storage System stores the data in a new partition and modifies the master volume's pointer to the new partition. The snapshot pointer does not change and remains pointing at the original data. The restoration process restores the pointer back to the original data and frees the modified partition space.

If a snapshot is taken and the original volume later increases in size, you can still do a restore operation. The snapshot still has the original volume size and will restore the original volume accordingly.

The XCLI session (or XCLI command) provides more options for restoration than the GUI. With the XCLI, you can restore a snapshot to a parent snapshot (Example 2-4).

Example 2-4 Restoring a snapshot to another snapshot

```
snapshot_restore snapshot=ITSO_Volume.snapshot_00002
target_snapshot=ITSO_Volume.snapshot_00001
```

2.2.5 Overwriting snapshots

For your regular backup jobs you can decide whether you always want to create new snapshots (and let the system delete the old ones) or whether you prefer to overwrite the existing snapshots with the latest changes to the data. For instance, a backup application requires the latest copy of the data to perform its backup operation. This overwrite operation modifies the pointers to the snapshot data to be reset to the master volume. Therefore, all pointers to the original data are lost, and the snapshot appears as new. Storage that was allocated for the data changes between the volume and its snapshot is released.

From either the Volumes and Snapshots view or the Snapshots Tree view, right-click the snapshot to overwrite. Select **Overwrite** from the menu and a dialog box opens. Click **OK** to validate the overwriting of the snapshot. Figure 2-22 illustrates overwriting the snapshot named `at12677_v3.snapshot_01`.

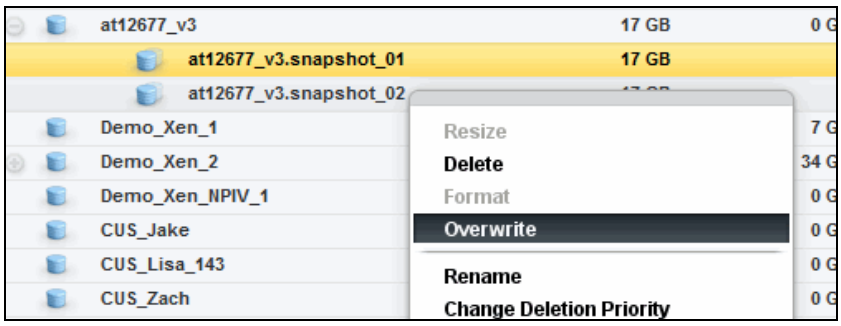


Figure 2-22 Overwriting a snapshot

An important note is that the overwrite process modifies the snapshot properties and pointers when involving duplicates. Figure 2-23 shows two changes to the properties. The snapshot named `at12677_v3.snapshot_01` has a new creation date. The duplicate snapshot still has the original creation date. However, it no longer points to the original snapshot. Instead, it points to the master volume according to the snapshot tree, which prevents a restoration of the duplicate to the original snapshot. If the overwrite occurs on the duplicate snapshot, the duplicate creation date is changed, and the duplicate is now pointing to the master volume.

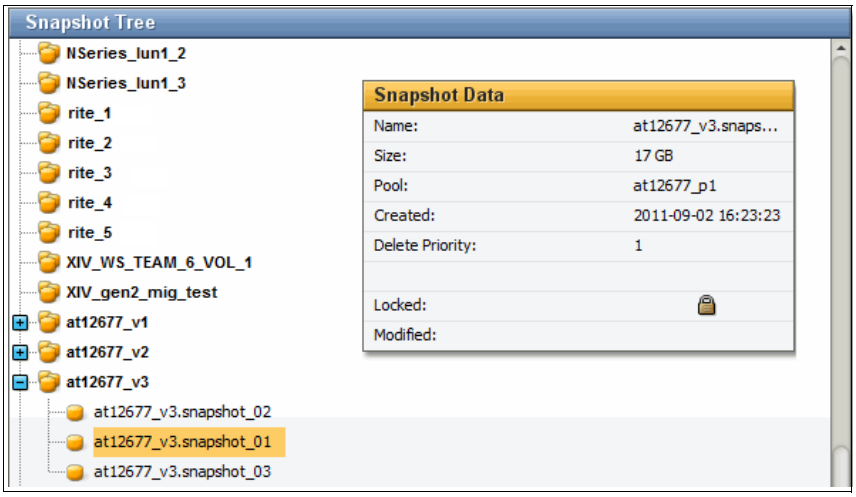


Figure 2-23 Snapshot tree after the overwrite process has occurred

The XCLI performs the overwrite operation through the **`snapshot_create`** command (Example 2-5). An optional parameter in the command specifies which snapshot to overwrite. If the optional parameter is not used, a new snapshot volume is created.

Example 2-5 Overwriting a snapshot

```
snapshot_create vol=ITSO_Volume overwrite=ITSO_Volume.snapshot_00001
```

2.2.6 Unlocking a snapshot

A snapshot also has the ability to be unlocked. By default, a snapshot is locked on creation and is only readable. Unlocking a snapshot allows the user to modify the data in the snapshot. This feature is useful for performing tests on a set of data or performing other types of data-mining activities.

Here are two scenarios that you must investigate when unlocking snapshots:

- The first scenario is to unlock a duplicate. By unlocking the duplicate, none of the snapshot properties are modified, and the structure remains the same. This method is straightforward and provides a backup of the master volume along with a working copy for modification. To unlock the snapshot, simply right-click the snapshot and select **Unlock**, as shown in Figure 2-24 on page 20.

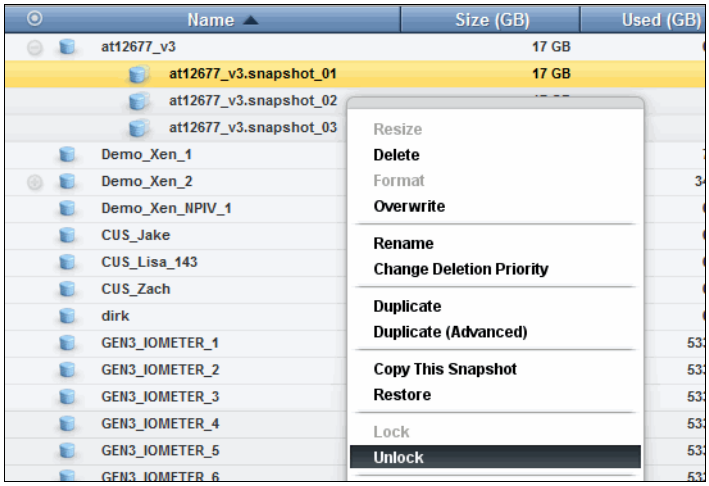


Figure 2-24 Unlocking a snapshot

The results in the **Snapshot Tree** window (Figure 2-25) show that the locked property is off and the modified property is on for `at12677_v3.snapshot_01`. Even if the volume is relocked or overwritten with the original master volume, the modified property remains on. Also note that in Figure 2-25 the structure is unchanged. If an error occurs in the modified duplicate snapshot, the duplicate snapshot can be deleted, and the original snapshot is duplicated a second time to restore the information.

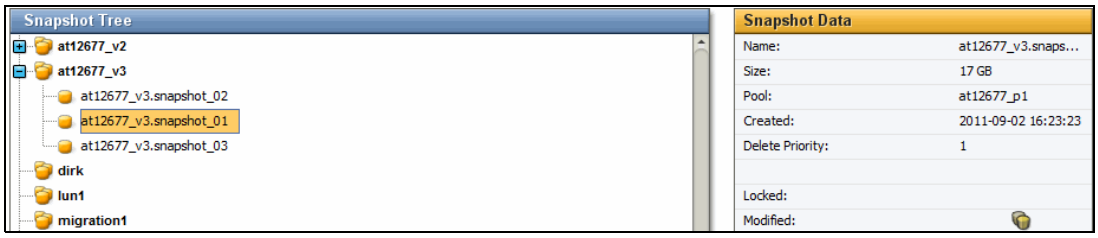


Figure 2-25 Unlocked duplicate snapshot

- For the second scenario, the original snapshot is unlocked and not the duplicate. Figure 2-26 shows the new property settings for `at12677_v3.snapshot_01`. At this point, the duplicate snapshot mirrors the unlocked snapshot, because both snapshots still point to the original data. While the unlocked snapshot is modified, the duplicate snapshot references the original data. If the unlocked snapshot is deleted, the duplicate snapshot remains, and its parent becomes the master volume.

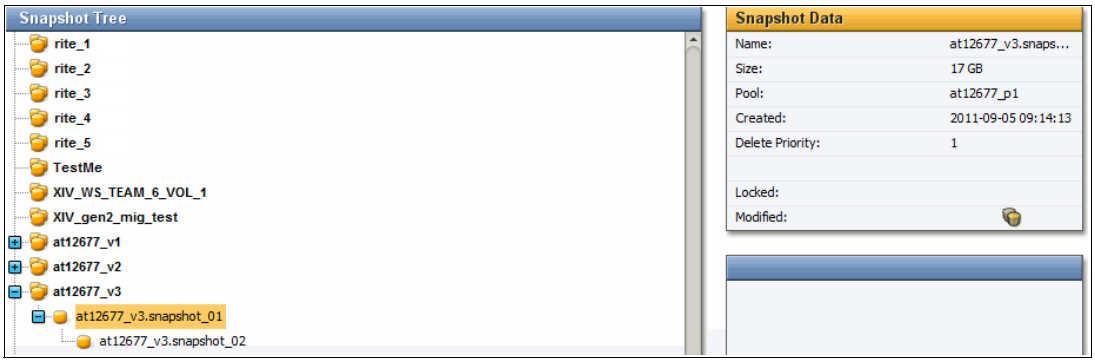


Figure 2-26 Unlocked original snapshot

Because the hierarchal snapshot structure was unmodified, the duplicate snapshot can be overwritten by the original snapshot. The duplicate snapshot can be restored to the master volume. Based on the results, this process does not differ from the first scenario. There is still a backup and a working copy of the data.

Example 2-6 shows the XCLI command to unlock a snapshot.

Example 2-6 Unlocking a snapshot with the XCLI session commands

```
vol_unlock vol=ITSO_Volume.snapshot_00001
```

2.2.7 Locking a snapshot

If the changes made to a snapshot must be preserved, you can lock an unlocked snapshot. Figure 2-27 shows locking the snapshot named `at12677_v3.snapshot_01`. From the Volumes and Snapshots panel, right-click the snapshot to lock and select **Lock**.

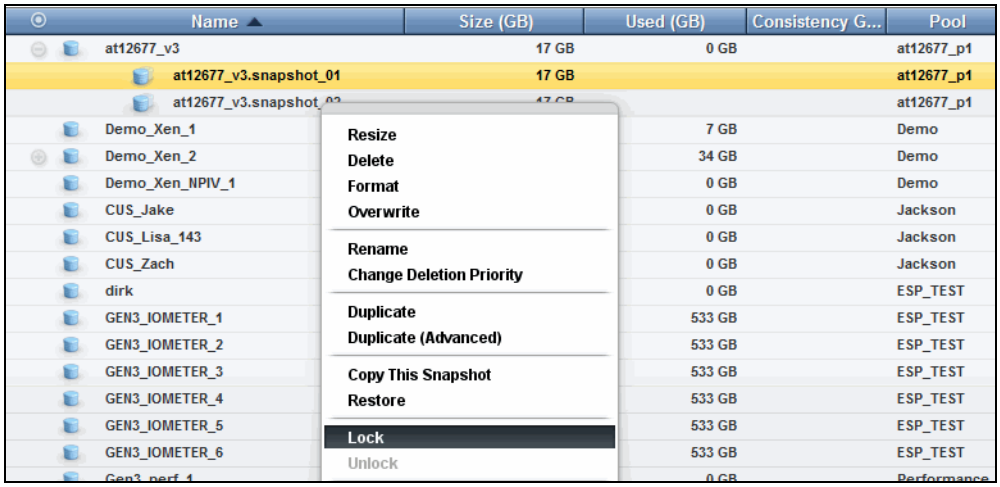


Figure 2-27 Locking a snapshot

The locking process completes immediately, preventing further modification to the snapshot. In Figure 2-28, the `at12677_v3.snapshot_01` snapshot shows that both the lock property and the modified property are on.

Even though there has not been a change to the snapshot, the system does not remove the modified property.

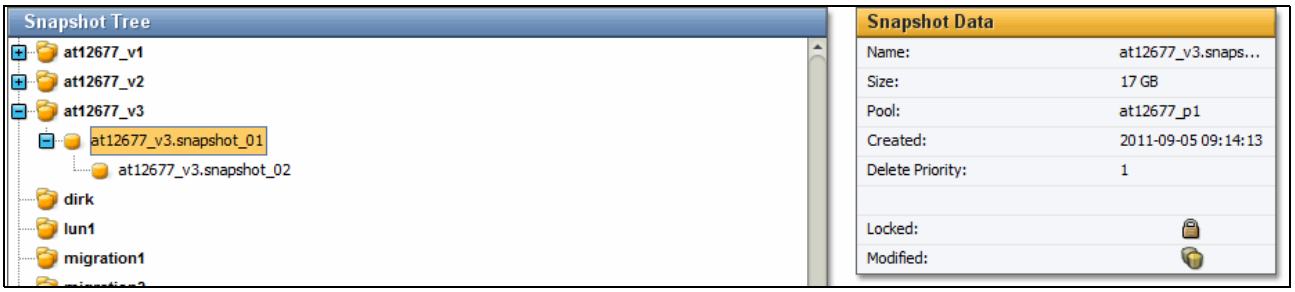


Figure 2-28 Validating that the snapshot is locked

The XCLI lock command (**vol_lock**), which is shown in Example 2-7, is similar to the unlock command. Only the actual command changes, but the same operating parameters are used when issuing the command.

Example 2-7 Locking a snapshot

```
vol_lock vol=ITSO_Volume.snapshot_00001
```

2.2.8 Deleting a snapshot

When a snapshot is no longer needed, you can delete it. Figure 2-29 illustrates how to delete a snapshot. In this case, the modified snapshot `at12677_v3.snapshot_01` is no longer needed. To delete the snapshot, right-click it and select **Delete** from the menu. A dialog box opens, requesting that you validate the operation.

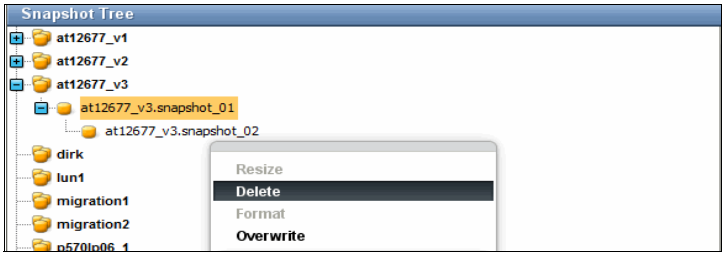


Figure 2-29 Deleting a snapshot

The panel in Figure 2-30 no longer displays the snapshot `at12677_v3.snapshot_01`. The volume and the duplicate snapshot are unaffected by the removal of this snapshot. In fact, the duplicate becomes the child of the master volume. The XIV Storage System provides the ability to restore the duplicate snapshot to the master volume or to overwrite the duplicate snapshot from the master volume even after deleting the original snapshot.

Figure 2-30 Validating the snapshot is removed

The delete snapshot command (**snapshot_delete**) operates the same as the creation snapshot. See Example 2-8.

Example 2-8 Deleting a snapshot

```
snapshot_delete snapshot=ITSO_Volume.snapshot_00001
```

Important: If you delete a volume, all snapshots associated with the volume are also deleted.

2.2.9 Automatic deletion of a snapshot

The XIV Storage System has a feature in place to protect a storage pool from becoming full. If the space allocated for snapshots becomes full, the XIV Storage System automatically deletes a snapshot. Figure 2-31 shows a storage pool with a 17 GB volume labeled `CSM_SMS`. The host connected to this volume is sequentially writing to a file that is stored on this volume. While the data is written, a snapshot called `CSM_SMS.snapshot_00001` is created, and one minute later, a second snapshot is taken (not a duplicate), which is called `CSM_SMS.snapshot_00002`.

Figure 2-31 Snapshot before the automatic deletion

With this scenario, a duplicate does not cause the automatic deletion to occur. Because a duplicate is a mirror copy of the original snapshot, the duplicate does not create the additional allocations in the storage pool.

Approximately one minute later, the oldest snapshot (`CSM_SMS.snapshot_00001`) is removed from the display. The storage pool is 51 GB in size, with a snapshot size of 34 GB, which is enough for one snapshot. If the master volume is unmodified, many snapshots can exist within the pool, and the automatic deletion does not occur. If there were two snapshots and two volumes, it might take longer to cause the deletion, because the volumes utilize different portions of the disks, and the snapshots might not have immediately overlapped.

To examine the details of the scenario at the point where the second snapshot is taken, a partition is in the process of being modified. The first snapshot caused a redirect on write, and a partition was allocated from the snapshot area in the storage pool. Because the second snapshot occurs at a different time, this action generates a second partition allocation in the storage pool space. This second allocation does not have available space, and the oldest snapshot is deleted. Figure 2-32 shows that the master volume `CSM_SMS` and the newest snapshot `CSM_SMS.snapshot_00002` are present. The oldest snapshot `CSM_SMS.snapshot_00001` was removed.

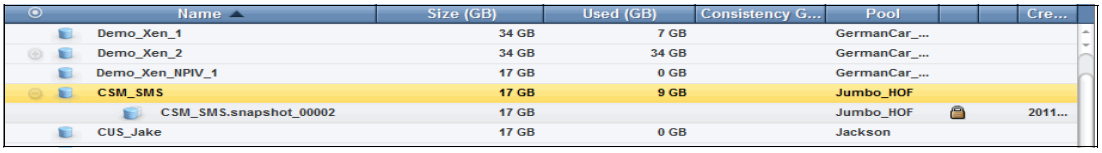


Figure 2-32 Snapshot after automatic deletion

To determine the cause of removal, go to the **Events** panel under the **Monitor** menu. As shown on Figure 2-33, the SNAPSHOT_DELETED_DUE_TO_POOL_EXHAUSTION event is logged.

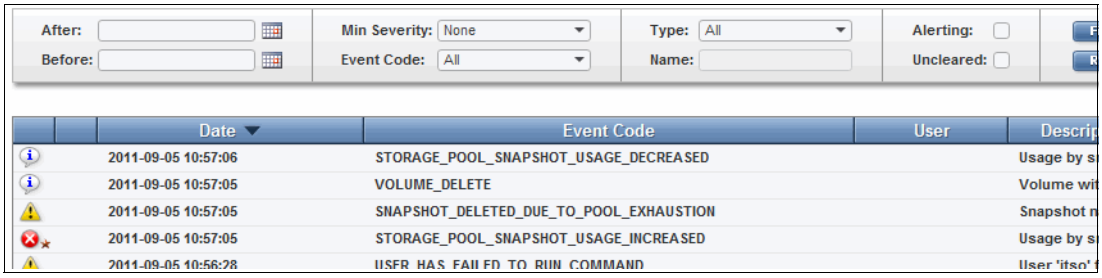


Figure 2-33 Record of automatic deletion

Selecting the Properties of the Volume_Delete Event Code will provide more information. The snapshot name CSM_SMS.snapshot_00001 and time stamp 2011-09-05 10:57:05 are logged for future reference (Figure 2-34).

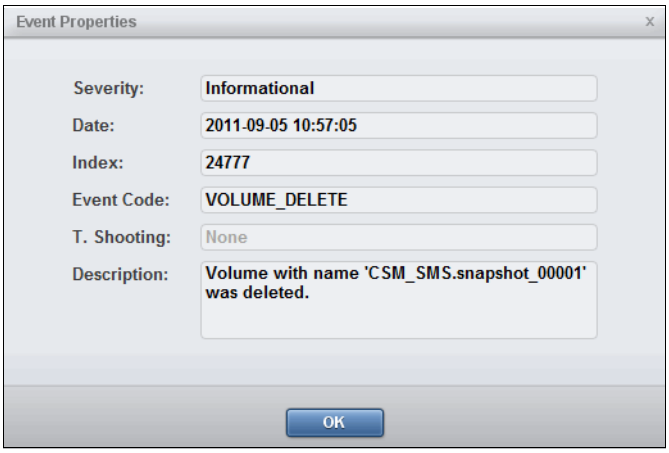


Figure 2-34 Volume_Delete Event Code Properties

2.3 Snapshots consistency group

A consistency group comprises multiple volumes so that a snapshot can be taken of all the volumes at the same moment in time. This action creates a synchronized snapshot of all the volumes and is ideal for applications that span multiple volumes, for example, a database application that stores its data files on multiple volumes. When creating a backup of the database, it is important to synchronize the data so that it is consistent.

2.3.1 Creating a consistency group

There are two methods of creating a consistency group:

- Create the consistency group and add the volumes in one step.
- Create the consistency group and then add the volumes in a subsequent step.

If you also use consistency groups to manage remote mirroring, you must first create an empty consistency group, mirror it, and later add mirrored volumes to the consistency group.

Restriction: Volumes in a consistency group must be in the same storage pool. A consistency group cannot include volumes from different pools.

Starting at the Volumes and Snapshots view, select the volume that is to be added to the consistency group. To select multiple volumes, hold down the Shift key or the Ctrl key to select/deselect individual volumes. After the volumes are selected, right-click a selected volume to open an operations menu. From there, click **Create a Consistency Group With Selected Volumes**. See Figure 2-35 for an example of this operation.

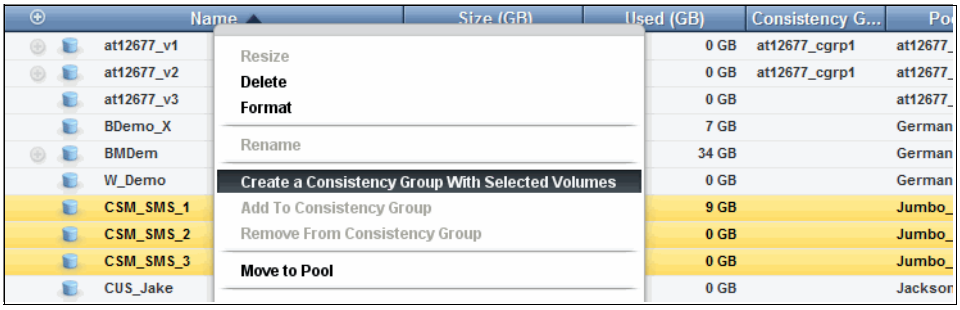


Figure 2-35 Creating a consistency group with selected volumes

After selecting the **Create** option from the menu, a dialog window opens. Enter the name of the consistency group. Because the volumes are added during creation, it is not possible to change the pool name. Figure 2-36 shows the process of creating a consistency group. After you enter the name, click **Create**.

Figure 2-36 Naming the consistency group

The volume consistency group ownership is visible under Volumes and Snapshots. As shown in Figure 2-37, the three volumes contained in the `Jumbo_HOF` pool are now owned by the `CSM_SMS_CG` consistency group. The volumes are displayed in alphabetical order and do not reflect a preference or internal ordering.

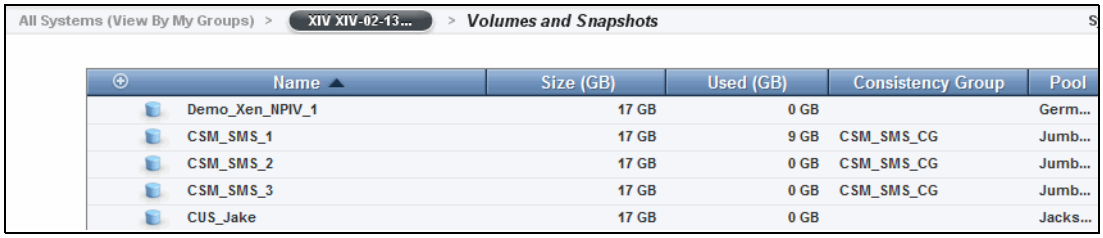


Figure 2-37 Viewing the volumes after creating a consistency group

To obtain details about the consistency group, the GUI provides a panel to view the information. From the **Volumes** menu, select **Consistency Groups**. Figure 2-38 illustrates how to access this panel.

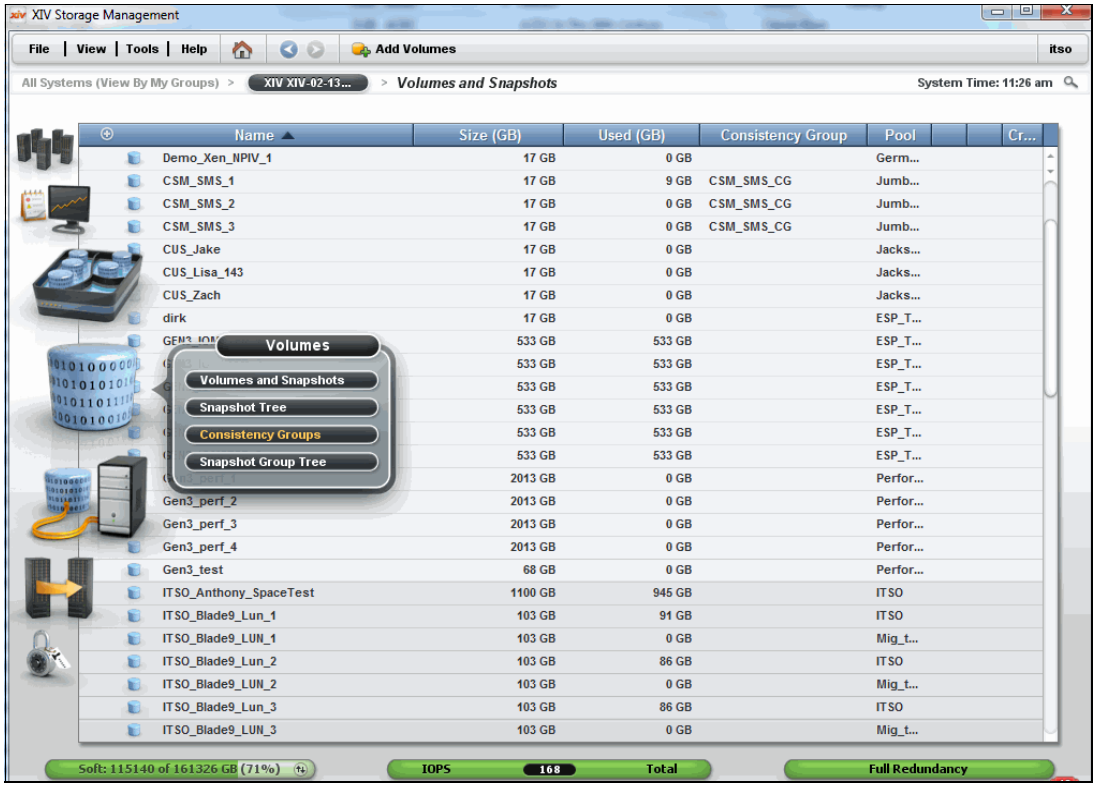

XIV Storage Management
File | View | Tools | Help
Add Volumes
itso
All Systems (View By My Groups) > XIV XIV-02-13... > Volumes and Snapshots
System Time: 11:26 am
Name
Size (GB)
Used (GB)
Consistency Group
Pool
Cr...
Demo_Xen_NPIV_1 17 GB 0 GB Germ...
CSM_SMS_1 17 GB 9 GB CSM_SMS_CG Jumb...
CSM_SMS_2 17 GB 0 GB CSM_SMS_CG Jumb...
CSM_SMS_3 17 GB 0 GB CSM_SMS_CG Jumb...
CUS_Jake 17 GB 0 GB Jacks...
CUS_Lisa_143 17 GB 0 GB Jacks...
CUS_Zach 17 GB 0 GB Jacks...
dirk 17 GB 0 GB ESP_T...
Volumes
Volumes and Snapshots
Snapshot Tree
Consistency Groups
Snapshot Group Tree
533 GB 533 GB ESP_T...
2013 GB 0 GB Perfor...
Gen3_perf_2 2013 GB 0 GB Perfor...
Gen3_perf_3 2013 GB 0 GB Perfor...
Gen3_perf_4 2013 GB 0 GB Perfor...
Gen3_test 68 GB 0 GB Perfor...
ITSO_Anthony_SpaceTest 1100 GB 945 GB ITSO
ITSO_Blade9_Lun_1 103 GB 91 GB ITSO
ITSO_Blade9_LUN_1 103 GB 0 GB Mig_t...
ITSO_Blade9_Lun_2 103 GB 86 GB ITSO
ITSO_Blade9_LUN_2 103 GB 0 GB Mig_t...
ITSO_Blade9_Lun_3 103 GB 86 GB ITSO
ITSO_Blade9_LUN_3 103 GB 0 GB Mig_t...
Soft: 115140 of 161326 GB (71%)
IOPS 168 Total
Full Redundancy

Figure 2-38 Accessing the consistency group view

This selection sorts the information by consistency group. The panel allows you to expand the consistency group and see all the volumes owned by that consistency group. In Figure 2-39, there are three volumes owned or contained by the `CSM_SMS_CG` consistency group. In this example, a snapshot of the volumes has not been created.

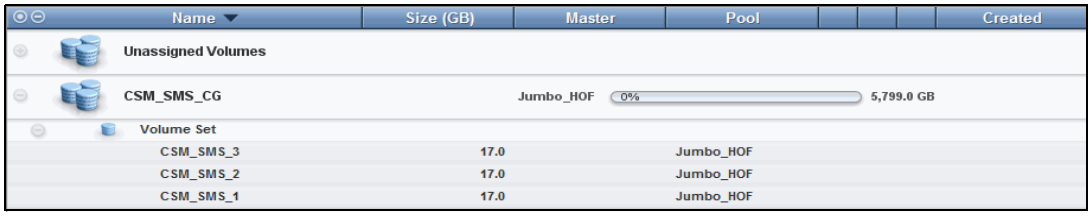


Figure 2-39 Consistency Groups view

From the consistency group view, you can create a consistency group without adding volumes. On the menu bar at the top of the window, there is an icon to add a new consistency group. Click the Create Consistency Group icon shown in Figure 2-40. A creation dialog box opens, as shown in Figure 2-36 on page 25. Then, provide a name and the storage pool for the consistency group.

Figure 2-40 Adding a new consistency group

When created, the consistency group opens in the **Consistency Groups** view of the GUI (Figure 2-41). The new group does not have any volumes associated with it. A new consistency group named `CSM_SMS_CG2` is created. The consistency group cannot be expanded yet because there are no volumes contained in the consistency group `CSM_SMS_CG2`.

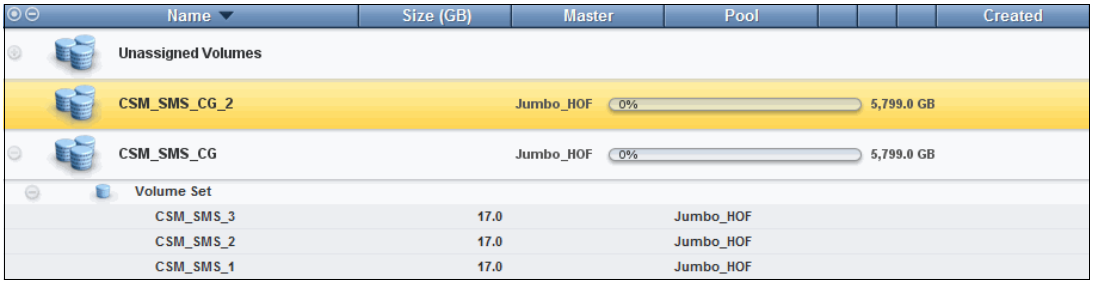


Figure 2-41 Validating new consistency group

Using the **Volumes** view in the GUI, select the volumes to add to the consistency group. After selecting the desired volumes, right-click the volumes and select **Add To Consistency Group**. Figure 2-42 shows three volumes being added to a consistency group:

- `CSM_SMS_4`
- `CSM_SMS_5`
- `CSM_SMS_6`

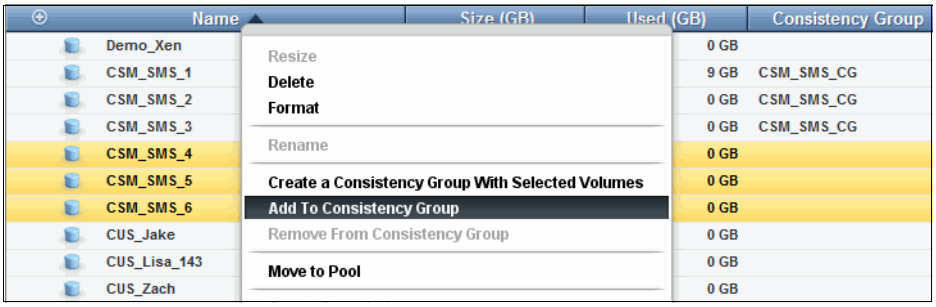


Figure 2-42 Adding volumes to a consistency group

After selecting the volumes to add, a dialog box opens asking for the consistency group to which to add the volumes. Figure 2-43 adds the volumes to the `CSM_SMS_CG` consistency group. Clicking **OK** completes the operation.

Figure 2-43 Selecting a consistency group for adding volumes

Using the XCLI session (or XCLI command), the process must be done in two steps. First, create the consistency group, then add the volumes. Example 2-9 provides an example of setting up a consistency group and adding volumes using the XCLI.

Example 2-9 Creating consistency groups and adding volumes with the XCLI

```
cg_create cg=ITSO_CG pool=itso
cg_add_vol cg=ITSO_CG vol=itso_volume_01
cg_add_vol cg=ITSO_CG vol=itso_volume_02
cg_add_vol cg=ITSO_CG vol=itso_volume_03
```

2.3.2 Creating a snapshot using consistency groups

When the consistency group is created and the volumes added, snapshots can be created. From the consistency group view on the GUI, select the consistency group to copy. As in Figure 2-44, right-click the group and select **Create Snapshot Group** from the menu. The system immediately creates a snapshot group.

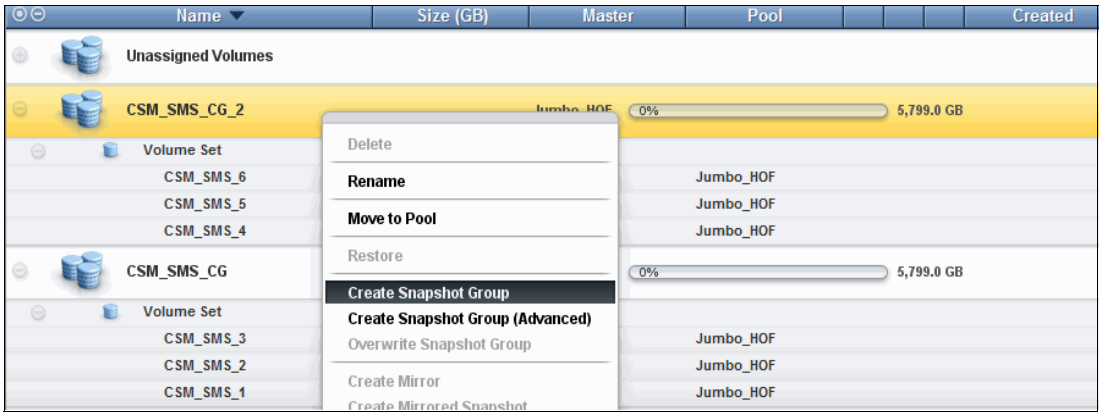


Figure 2-44 Creating a snapshot using consistency groups

The new snapshots are created and displayed beneath the volumes in the **Consistency Groups** view (Figure 2-45). These snapshots have the same creation date and time. Each snapshot is locked on creation and has the same defaults as a regular snapshot. The snapshots are contained in a group structure (called a *snapshot group*) that allows all the snapshots to be managed by a single operation.

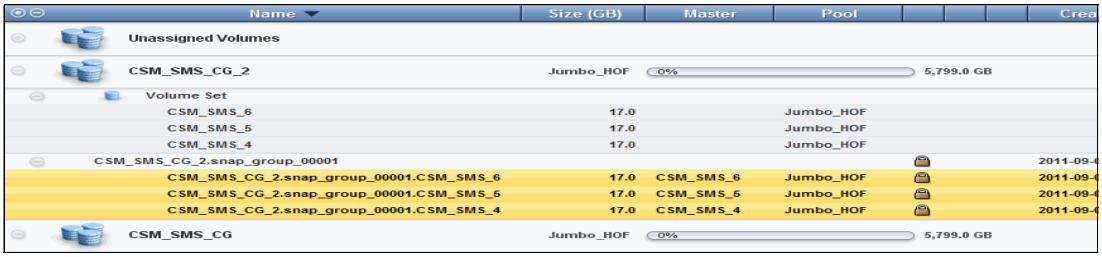


Figure 2-45 Validating the new snapshots in the consistency group

Adding volumes to a consistency group does not prevent you from creating a single volume snapshot. If a single volume snapshot is created, it is not displayed in the consistency group view. The single volume snapshot is also not consistent across multiple volumes. However, the single volume snapshot does work according to all the rules defined previously in 2.2, ".Snapshot handling" on page 11.

With the XCLI, when the consistency group is set up, it is simple to create the snapshot. One command creates all the snapshots within the group at the same moment (Example 2-10).

Example 2-10 Creating a snapshot group

```
cg_snapshots_create cg=ITSO_CG
```

2.3.3 Managing a consistency group

After the snapshots are created within a consistency group, you have several options available. The same management options for a snapshot are available to a consistency group. Specifically, the deletion priority is modifiable, and the snapshot or group can be unlocked and locked, and the group can be restored or overwritten. See 2.2, ".Snapshot handling" on page 11, for specific details about performing these operations.

In addition to the snapshot functions, you can remove a volume from the consistency group. By right-clicking the volume, a menu opens. Click **Remove From Consistency Group** and validate the removal on the dialog window that opens. Figure 2-46 provides an example of removing the `CSM_SMS_6` volume from the consistency group.

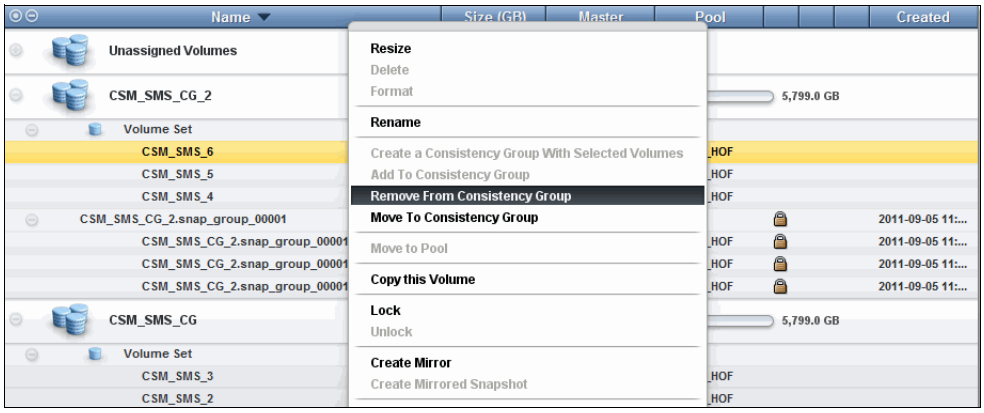


Figure 2-46 Removing a volume from a consistency group

Removing a volume from a consistency group after a snapshot is performed prevents restoration of any snapshots in the group. If the volume is added back into the group, the group can be restored.

To obtain details about a consistency group, you can select **Snapshots Group Tree** from the **Volumes** menu. Figure 2-47 shows where to find the group view.

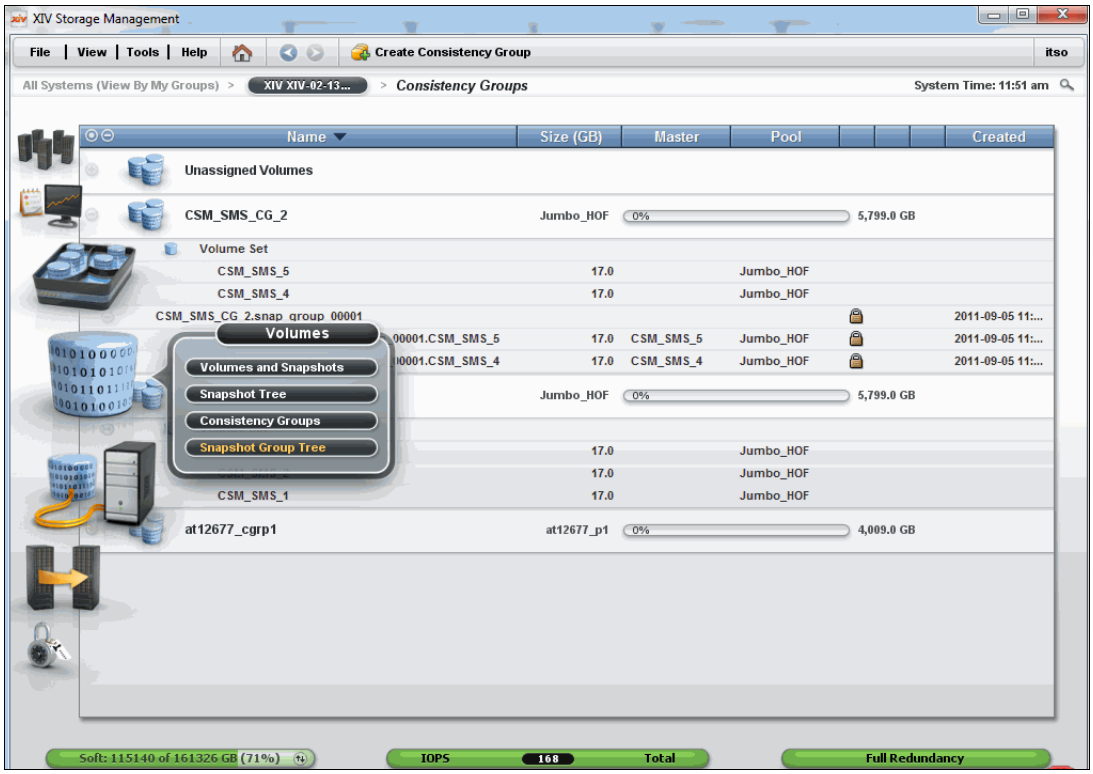


Figure 2-47 Selecting the Snapshot Group Tree

From the `Snapshots Group Tree` view, you can see many details. Select the group to view on the left panel by clicking the group snapshot. The right-side panes provide more in-depth information about the creation time, the associated pool, and the size of the snapshots. In addition, the consistency group view points out the individual snapshots present in the group. See Figure 2-48 for an example of the data that is contained in a consistency group.

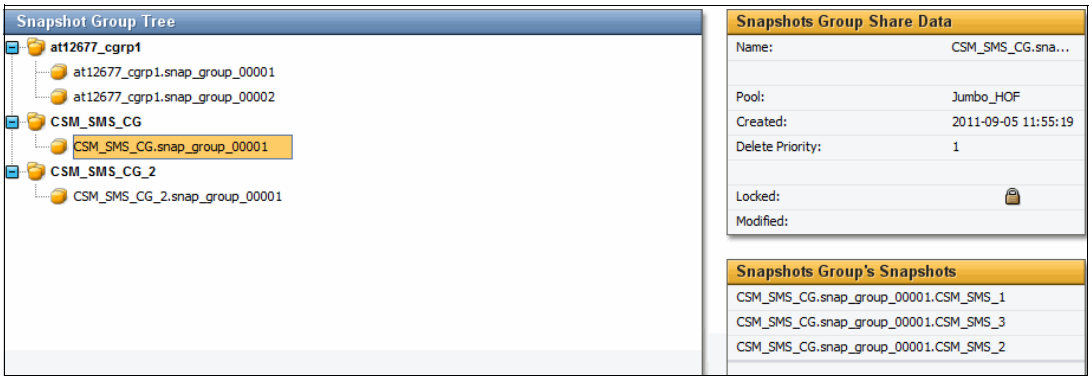


Figure 2-48 Snapshots Group Tree view

To display all the consistency groups in the system, issue the XCLI **cg_list** command (Example 2-11).

Example 2-11 Listing the consistency groups

```
cg_list

Name              Pool Name
itso_esx_cg       itso
itso_mirror_cg    itso
nn_cg_residency   Residency_nils
db2_cg            itso
sync_rm           1_Sales_Pool
ITSO_i_Mirror     ITSO_IBM_i
itso_srm_cg       ITSO_SRM
Team01_CG         Team01_RP
ITSO_CG           itso
ITSO_CG2          itso
```

More details are available by viewing all the consistency groups within the system that have snapshots. The groups can be unlocked or locked, restored, or overwritten. All the operations discussed in the snapshot section are available with the **snap_group** operations.

Example 2-12 illustrates the **snap_group_list** command.

Example 2-12 Listing all the consistency groups with snapshots

```
snap_group_list
                                                                  Deletion
Name                              CG              Snapshot Time   Priority
db2_cg.snap_group_00001           db2_cg          2010-09-30 13:26:21   1
ITSO_CG.snap_group_00001          ITSO_CG         2010-10-12 11:24:54   1
ITSO_CG.snap_group_00002          ITSO_CG         2010-10-12 11:44:02   1
last-replicated-ITSO_i_Mirror     ITSO_i_Mirror   2010-10-12 13:21:41   1
most-recent-ITSO_i_Mirror         ITSO_i_Mirror   2010-10-12 13:22:00   1
```

2.3.4 Deleting a consistency group

Before a consistency group can be deleted, the associated volumes must be removed from the consistency group. On deletion of a consistency group, the snapshots become independent snapshots and remain tied to their volume. To delete the consistency group, right-click the group and select **Delete**. Validate the operation by clicking **OK**. Figure 2-49 provides an example of deleting the consistency group called `CSM_SMS_CG2`.

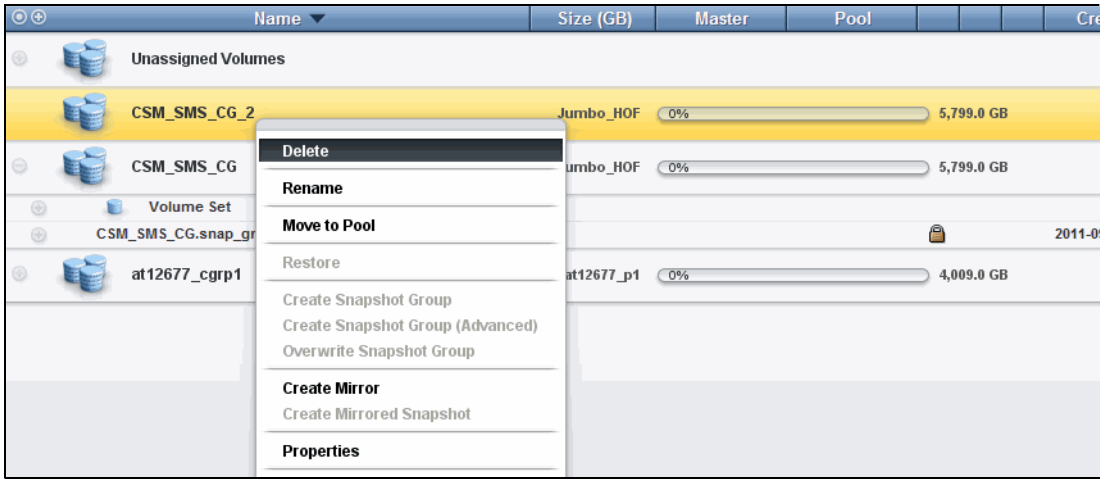


Figure 2-49 Deleting a consistency group

To delete a consistency group with the XCLI, first remove all the volumes one at a time. As in Example 2-13, each volume in the consistency group is removed first. Then the consistency group is available for deletion. Deletion of the consistency group does not delete the individual snapshots. They are tied to the volumes and are removed from the consistency group when you remove the volumes.

Example 2-13 Deleting a consistency group

```
cg_remove_vol vol=itso_volume_1
cg_remove_vol vol=itso_volume_2
cg_remove_vol vol=isto_volume_3
cg_delete cg=ITSO_CG
```

2.4 Snapshot with remote mirror

XIV has a special snapshot (shown in Figure 2-50) that is automatically created by the system. During the recovery phase of a remote mirror, the system creates a snapshot on the target to ensure a consistent copy.

Important: This snapshot has a special deletion priority and is not deleted automatically if the snapshot space becomes fully utilized.

When the synchronization is complete, the snapshot is removed by the system because it is no longer needed. The following list describes the sequence of events to trigger the creation of the special snapshot. If a write does not occur while the links are broken, the system does not create the special snapshot. The events are as follows:

1. Remote mirror is synchronized.
2. Loss of connectivity to remote system occurs.
3. Writes continue to the primary XIV Storage System.
4. Mirror paths are re-established (here the snapshot is created) and synchronization starts.

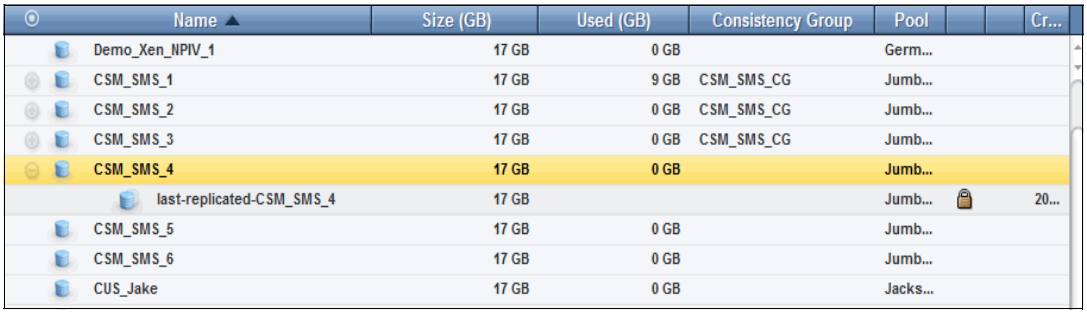

Name
Size (GB)
Used (GB)
Consistency Group
Pool
Cr...
Demo_Xen_NPIV_1 17 GB 0 GB Germ...
CSM_SMS_1 17 GB 9 GB CSM_SMS_CG Jumb...
CSM_SMS_2 17 GB 0 GB CSM_SMS_CG Jumb...
CSM_SMS_3 17 GB 0 GB CSM_SMS_CG Jumb...
CSM_SMS_4 17 GB 0 GB Jumb...
last-replicated-CSM_SMS_4 17 GB Jumb... 20...
CSM_SMS_5 17 GB 0 GB Jumb...
CSM_SMS_6 17 GB 0 GB Jumb...
CUS_Jake 17 GB 0 GB Jacks...

Figure 2-50 Special snapshot during remote mirror synchronization operation

For more details about remote mirror see Chapter 5, “Synchronous Remote Mirroring” on page 113.

Important: The special snapshot is created regardless of the amount of pool space on the target pool. If the snapshot causes the pool to be overutilized, the mirror remains inactive. The pool must be expanded to accommodate the snapshot, then the mirror can be re-established.

2.5 MySQL database backup example

MySQL is an open source database application that is used by many web programs. For more information go to the following location:

http://www.mysql.com

The database has several important files:

- The database data
- The log data
- The backup data

The MySQL database stores the data in a set directory and cannot be separated. The backup data, when captured, can be moved to a separate system. The following scenario shows an incremental backup of a database and then uses snapshots to restore the database to verify that the database is valid.

The first step is to back up the database. For simplicity, a script is created to perform the backup and take the snapshot. Two volumes are assigned to a Linux host (Figure 2-51). The first volume contains the database and the second volume holds the incremental backups in case of a failure.

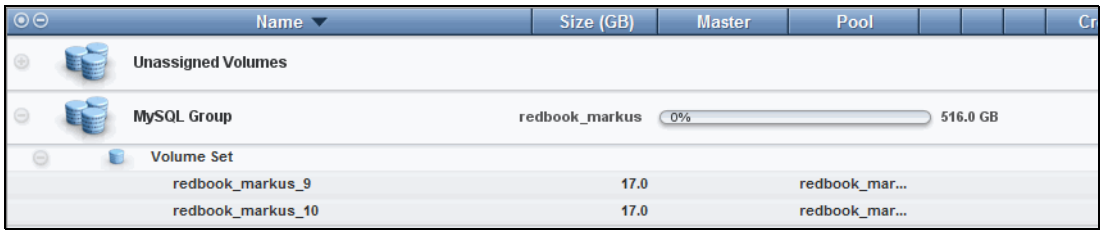


Figure 2-51 XIV view of the volumes

On the Linux host, the two volumes are mapped onto separate file systems. The first file system `xiv_pfe_1` maps to volume `redbook_markus_09`, and the second file system `xiv_pfe_2` maps to volume `redbook_markus_10`. These volumes belong to the consistency group `MySQL Group` so that when the snapshot is taken, snapshots of both volumes are taken at the same moment.

To do the backup, you must configure the following items:

- The XIV XCLI must be installed on the server. This way, the backup script can invoke the snapshot instead of relying on human intervention.
- Secondly, the database must have the incremental backups enabled. To enable the incremental backup feature, MySQL must be started with the --log-bin feature (Example 2-14). This feature enables the binary logging and allows database restorations.

Example 2-14 Starting MySQL

```
./bin/mysqld_safe --no-defaults --log-bin=backup
```

The database is installed on /xiv_pfe_1. However, a pointer in */usr/local* is made, which allows all the default settings to coexist, and yet the database is stored on the XIV volume. To create the pointer, use the command in Example 2-15. The source directory must be changed for your particular installation. You can also install the MySQL application on a local disk and change the default data directory to be on the XIV volume.

Example 2-15 MySQL setup

```
cd /usr/local
ln -s /xiv_pfe_1/mysql-5.0.51a-linux-i686-glibc23 mysql
```

The backup script is simple, and depending on the implementation of your database, the following script might be too simple. However, the following script (Example 2-16) does force an incremental backup and copies the data to the second XIV volume. Then the script locks the tables so that no more data can be modified. When the tables are locked, the script initiates a snapshot, which saves everything for later use. Finally, the tables are unlocked.

Example 2-16 Script to perform backup

```
# Report the time of backing up
date

# First flush the tables this can be done while running and
# creates an incremental backup of the DB at a set point in time.
/usr/local/mysql/bin/mysql -h localhost -u root -p password < ~/SQL_BACKUP

# Since the mysql daemon was run specifying the binary log name
# of backup the files can be copied to the backup directory on another disk
cp /usr/local/mysql/data/backup* /xiv_pfe_2

# Secondly lock the tables so a Snapshot can be performed.
/usr/local/mysql/bin/mysql -h localhost -u root -p password < ~/SQL_LOCK

# XCLI command to perform the backup
# ****** NOTE User ID and Password are set in the user profile *****
/root/XIVGUI/xcli -c xiv_pfe cg_Snapshots_create cg="MySQL Group"

# Unlock the tables so that the database can continue in operation.
/usr/local/mysql/bin/mysql -h localhost -u root -p password < ~/SQL_UNLOCK
```

When issuing commands to the MySQL database, the password for the root user is stored in an environment variable (not in the script, as was done in Example 2-16 for simplicity). Storing the password in an environment variable allows the script to perform the action without requiring user intervention. For the script to invoke the MySQL database, the SQL statements are stored in separate files and piped into the MySQL application. Example 2-17 provides the three SQL statements that are issued to perform the backup operation.

Example 2-17 SQL commands to perform backup operation

```
SQL_BACKUP
   FLUSH TABLES

SQL_LOCK
   FLUSH TABLES WITH READ LOCK

SQL_UNLOCK
   UNLOCK TABLES
```

Before running the backup script, a test database, which is called `redbook`, is created. The database has one table, which is called `chapter`, which contains the chapter name, author, and pages. The table has two rows of data that define information about the chapters in the database. Figure 2-52 shows the information in the table before the backup is performed.

```
mysql> SHOW TABLES;
+------------------+
| Tables_in_redbook |
+------------------+
| chapter           |
+------------------+
1 row in set (0.00 sec)

mysql> SELECT * FROM chapter;
+-------------+--------+-------+
| name        | author | pages |
+-------------+--------+-------+
| config      | Atilla |   200 |
| performance | Chris  |   100 |
+-------------+--------+-------+
2 rows in set (0.00 sec)

mysql>
```

Figure 2-52 Data in database before backup

Now that the database is ready, the backup script is run. Example 2-18 is the output from the script. Then the snapshots are displayed to show that the system now contains a backup of the data.

Example 2-18 Output from the backup process

```
[root@x345-tic-30 ~]# ./mysql_backup

Mon Aug 11 09:12:21 CEST 2008
Command executed successfully.

[root@x345-tic-30 ~]# /root/XIVGUI/xcli -c xiv_pfe snap_group_list cg="MySQLGroup"
Name                          CG           Snapshot Time        Deletion Priority
MySQL Group.snap_group_00006  MySQL Group  2008-08-11 15:14:24  1

[root@x345-tic-30 ~]# /root/XIVGUI/xcli -c xiv_pfe time_list
Time      Date        Time Zone      Daylight Saving Time
15:17:04  2008-08-11  Europe/Berlin  yes
[root@x345-tic-30 ~]#
```

To show that the restore operation is working, the database is dropped (Figure 2-53) and all the data is lost. After the drop operation is complete, the database is permanently removed from MySQL. It is possible to perform a restore action from the incremental backup. For this example, the snapshot function is used to restore the entire database.

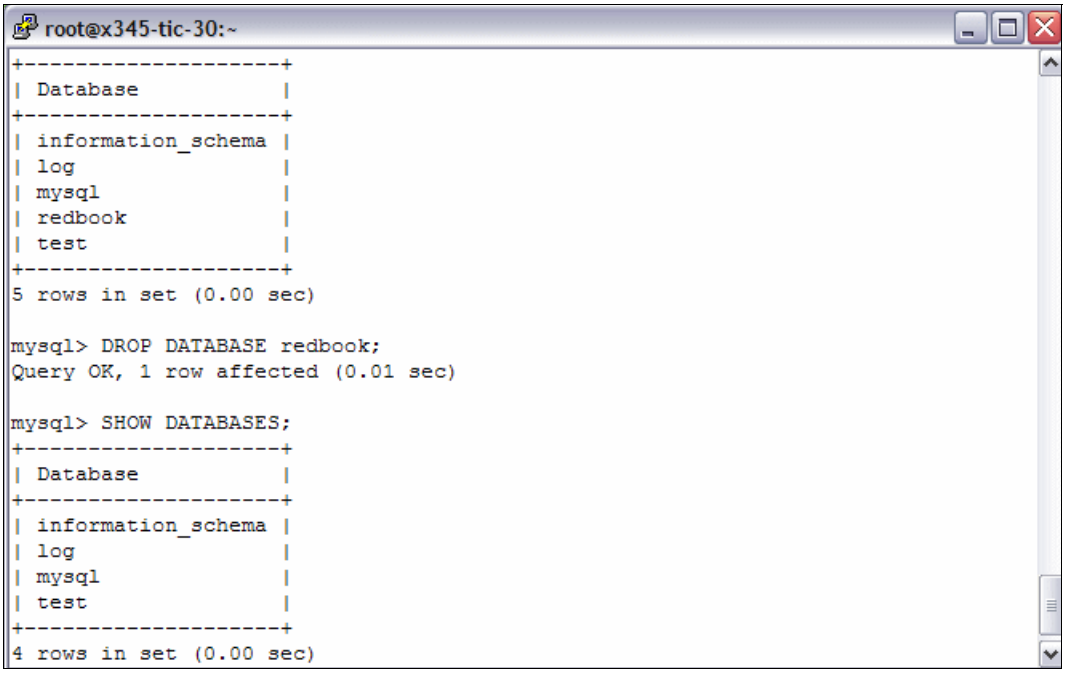


Figure 2-53 Dropping the database

The restore script (Example 2-19) stops the MySQL daemon and unmounts the Linux file systems. Then the script restores the snapshot and finally remounts and starts MySQL.

Example 2-19 Restore script

```
[root@x345-tic-30 ~]# cat mysql_restore
# This resotration just overwrites all in the database and puts the
# data back to when the snapshot was taken. It is also possible to do
# a restore based on the incremental data; this script does not handle
# that condition.

# Report the time of backing up
date

# First shutdown mysql
mysqladmin -u root -p password shutdown

# Unmount the filesystems
umount /xiv_pfe_1
umount /xiv_pfe_2

#List all the snap groups
/root/XIVGUI/xcli -c xiv_pfe snap_group_list cg="MySQL Group"

#Prompt for the group to restore
echo "Enter Snapshot group to restore: "
read -e snap_group
```

```
# XCLI command to perform the backup
# ****** NOTE User ID and Password are set in the user profile *****
/root/XIVGUI/xcli -c xiv_pfe snap_group_restore snap_group="$snap_group"

# Mount the FS
mount /dev/dm-2 /xiv_pfe_1
mount /dev/dm-3 /xiv_pfe_2

# Start the MySQL server
cd /usr/local/mysql
./configure
```

Example 2-20 shows the output from the restore action.

Example 2-20 Output from the restore script

```
[root@x345-tic-30 ~]# ./mysql_restore
Mon Aug 11 09:27:31 CEST 2008
STOPPING server from pid file
/usr/local/mysql/data/x345-tic-30.mainz.de.ibm.com.pid
080811 09:27:33  mysqld ended

Name                          CG           Snapshot Time        Deletion Priority
MySQL Group.snap_group_00006  MySQL Group  2008-08-11 15:14:24  1
Enter Snapshot group to restore:
MySQL Group.snap_group_00006
Command executed successfully.
NOTE: This is a MySQL binary distribution. It's ready to run, you don't
need to configure it!

To help you a bit, I am now going to create the needed MySQL databases
and start the MySQL server for you.  If you run into any trouble, please
consult the MySQL manual, that you can find in the Docs directory.

Installing MySQL system tables...
OK
Filling help tables...
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:
./bin/mysqladmin -u root password 'new-password'
./bin/mysqladmin -u root -h x345-tic-30.mainz.de.ibm.com password 'new-password'

Alternatively you can run:
./bin/mysql_secure_installation

which also gives the option of removing the test
databases and anonymous user created by default.  This is
strongly recommended for production servers.

See the manual for more instructions.
```

```
You can start the MySQL daemon with:
cd . ; ./bin/mysqld_safe &

You can test the MySQL daemon with mysql-test-run.pl
cd mysql-test ; perl mysql-test-run.pl

Please report any problems with the ./bin/mysqlbug script!

The latest information about MySQL is available on the Web at
http://www.mysql.com
Support MySQL by buying support/licenses at http://shop.mysql.com
Starting the mysqld server.  You can test that it is up and running
with the command:
./bin/mysqladmin version
[root@x345-tic-30 ~]# Starting mysqld daemon with databases from
/usr/local/mysql/data
```

When complete, the data is restored and the database is available, as shown in Figure 2-54.

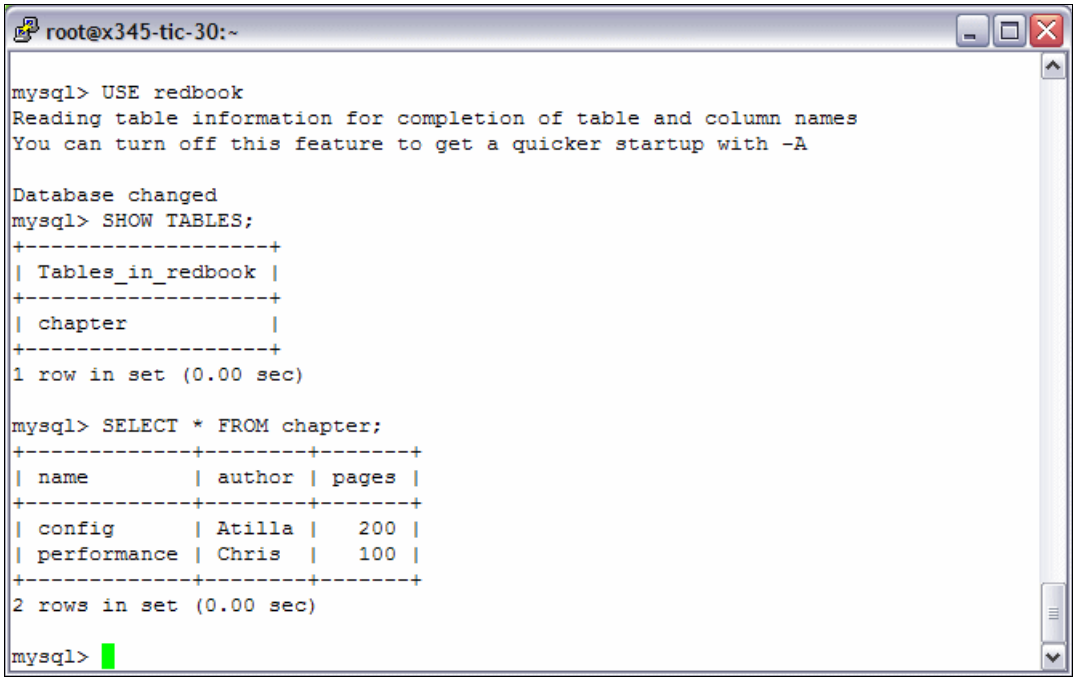

root@x345-tic-30:~
mysql> USE redbook
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
mysql> SHOW TABLES;
+-------------------+
| Tables_in_redbook |
+-------------------+
| chapter |
+-------------------+
1 row in set (0.00 sec)
mysql> SELECT * FROM chapter;
+-------------+--------+-------+
| name | author | pages |
+-------------+--------+-------+
| config | Atilla | 200 |
| performance | Chris | 100 |
+-------------+--------+-------+
2 rows in set (0.00 sec)
mysql>

Figure 2-54 Database after restore operation

2.6 Snapshot example for a DB2 database

Guidelines and recommendations of how to use the IBM XIV Storage System in database application environments are in k *IBM XIV Storage System: Host Attachment and Interoperability*, SG24-7904.

The following example scenario illustrates how to prepare an IBM DB2® database on an AIX platform for storage-based snapshot backup and then perform snapshot backup and restore.

IBM offers the Tivoli Storage FlashCopy® Manager software product to automate creation and restore of consistent database snapshot backups and to offload the data from the snapshot backups to an external backup/restore system like Tivoli Storage Manager. The previously mentioned book includes an overview chapter about Tivoli Storage FlashCopy Manager. For more details, visit the following locations:

- http://www.ibm.com/software/tivoli/products/storage-flashcopy-mgr
- http://publib.boulder.ibm.com/infocenter/tsminfo/v6

2.6.1 XIV Storage System and AIX OS environments

In this example, the database is named `XIV` and is stored in the file system `/db2/XIV/db2xiv`. The file system `/db2/XIV/log_dir` is intended to be used for the database log files. Figure 2-55 and Example 2-21 show the XIV volumes and the AIX file systems that were created for the database.

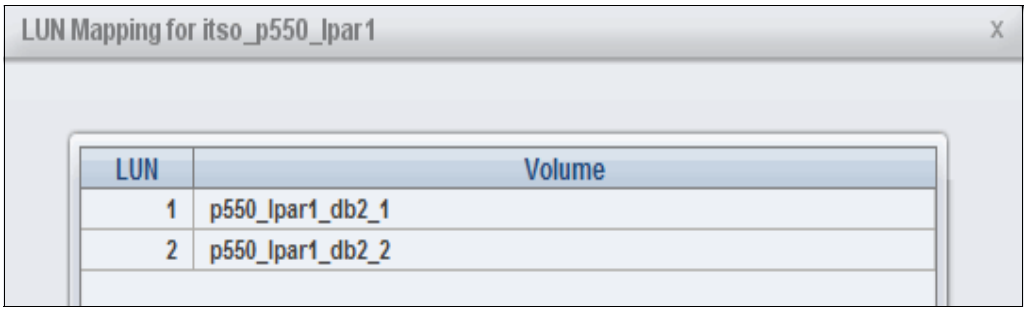


Figure 2-55 XIV volume mapping for the DB2 database server

Example 2-21 AIX volume groups and file systems created for the DB2 database

```
$ lsvg
rootvg
db2datavg
db2logvg
$ df -g
Filesystem    GB blocks      Free %Used    Iused %Iused Mounted on
/dev/hd4           2.31      0.58   75%    19508    12% /
/dev/hd2           1.75      0.14   92%    38377    46% /usr
/dev/hd9var        0.16      0.08   46%     4573    19% /var
/dev/hd3           5.06      2.04   60%     7418     2% /tmp
/dev/hd1           1.00      0.53   48%       26     1% /home
/dev/hd11admin      0.12      0.12    1%        5     1% /admin
/proc                 -         -    -         -     -  /proc
/dev/hd10opt       1.69      1.52   10%     2712     1% /opt
/dev/livedump      0.25      0.25    1%        4     1% /var/adm/ras/livedump
/dev/db2loglv     47.50     47.49    1%        4     1% /db2/XIV/log_dir
/dev/db2datalv     47.50     47.31    1%       56     1% /db2/XIV/db2xiv
```

2.6.2 Preparing the database for recovery

All databases have logs associated with them. These logs keep records of database changes. When a DB2 database is created, *circular* logging is the default behavior, which means DB2 uses a set of transaction log files in round-robin mode. With this type of logging, only full, offline backups of the database are allowed. To perform an online backup of the database, the logging method must be changed to *archive*. See Example 2-22.

This DB2 configuration change enables consistent XIV snapshot creation of the XIV volumes (that the database is stored on) while the database is online, restore of the database using snapshots, and roll forward of the database changes to a desired point in time.

Connect to DB2 as a database administrator to change the database configuration.

Example 2-22 Changing DB2 logging method

```
$ db2 connect to XIV

Database Connection Information
 Database server        = DB2/AIX64 9.7.0
 SQL authorization ID   = DB2XIV
 Local database alias   = XIV

$ db2 update db cfg using LOGARCHMETH1 LOGRETAIN
$ db2 update db cfg using NEWLOGPATH /db2/XIV/log_dir
```

After the archive logging method has been enabled, DB2 requests a database backup.

Example 2-23

```
$ db2 connect reset
$ db2 backup db XIV to /tmp
$ db2 connect to XIV
```

Before the snapshot creation, ensure that the snapshot includes all file systems relevant for the database backup. If in doubt, the *dbpath* view shows this information (Example 2-24). The output only shows the relevant lines for better readability.

Example 2-24 DB2 dbpath view

```
$ db2 select path from sysibmadm.dbpaths
/db2/XIV/log_dir/NODE0000/
/db2/XIV/db2xiv/
/db2/XIV/db2xiv/db2xiv/NODE0000/sqldbdir/
/db2/XIV/db2xiv/db2xiv/NODE0000/SQL00001/
```

The AIX commands **df** and **lsvg** (with the **-l** and **-p** options) identify the related AIX file systems and device files (hdisks). The XIV utility **xiv_devlist** shows the AIX hdisk names and the names of the associated XIV volumes.

2.6.3 Using XIV snapshots for database backup

The following procedure creates a snapshot of a primary database for use as a backup image. This procedure can be used instead of performing backup database operations on the primary database.

1. Suspend write I/O on the database.

   ```
   $ db2 set write suspend for database
   ```

2. Create XIV snapshots.

   While the database I/O is suspended, generate a snapshot of the XIV volumes the database is stored on. A snapshot of the log file is not created. This makes it possible to recover to a certain point in time instead of just going back to the last consistent snapshot image after database corruption occurs.

   Example 2-25 shows the xcli commands to create a consistent snapshot.

   Example 2-25 XCLI commands to create a consistent XIV snapshot

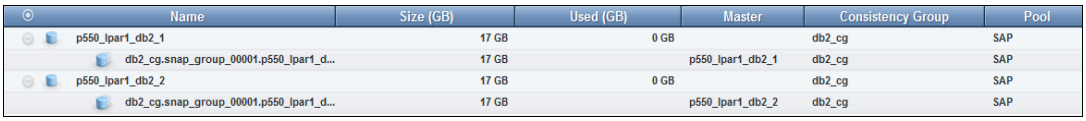
   ```
   XIV LAB 3 1300203>>cg_create cg=db2_cg pool=itso
   Command executed successfully.
   XIV LAB 3 1300203>>cg_add_vol vol=p550_lpar1_db2_1 cg=db2_cg
   Command executed successfully.
   XIV LAB 3 1300203>>cg_snapshots_create cg=db2_cg
   Command executed successfully.
   ```

3. Resume database write I/O.

   After the snapshot has been created, database write I/O can be resumed.

   ```
   $ db2 set write resume for db
   ```

Figure 2-56 shows the newly created snapshot on the XIV graphical user interface.



Name	Size (GB)	Used (GB)	Master	Consistency Group	Pool
p550_lpar1_db2_1	17 GB	0 GB		db2_cg	SAP
db2_cg.snap_group_00001.p550_lpar1_d...	17 GB		p550_lpar1_db2_1	db2_cg	SAP
p550_lpar1_db2_2	17 GB	0 GB		db2_cg	SAP
db2_cg.snap_group_00001.p550_lpar1_d...	17 GB		p550_lpar1_db2_2	db2_cg	SAP

Figure 2-56 XIV snapshot of the DB2 database volume

2.6.4 Restoring the database from the XIV snapshot

If a failure occurs on the primary system, or data is corrupted requiring a restore from backup, use the following steps to bring the database to the state before the corruption occurred. In a production environment a forward recovery to a certain point in time might be required. In this case the DB2 recover command requires other options, but the following process to handle XIV storage system and operating system is still valid.

1. Terminate database connections and stop the database.

   ```
   $ db2 connect reset
   $ db2stop
   ```

2. On the AIX system un-mount the file systems the database resides in and deactivate the volume groups.

   ```
   # umount /db2/XIV/db2xiv
   # varyoffvg db2datavg
   ```

3. Restore the data volumes from the XIV snapshot (Example 2-26).

Example 2-26 CLI command to restore a XIV snapshot

```
XIV LAB 3 1300203>>snap_group_restore snap_group=db2_cg.snap_group_00001

Warning:   ARE_YOU_SURE_YOU_WANT_TO_RESTORE_SNAPGROUP y/n:
Command executed successfully.
```

4. On the AIX system activate the volume groups and mount the file systems the database resides in.

```
# varyonvg db2datavg
# mount /db2/XIV/db2xiv
```

5. Start the database instance.

```
$ db2start
```

6. Initialize the database.

   From the DB2 view the XIV snapshot of the database volumes creates a split mirror database environment. The database was in write suspend mode when the snapshot was taken. Thus the restored database is still in this state and the split mirror must be used as a backup image to restore the primary database. The DB2 command **db2inidb** must to run to initialize a mirrored database before the split mirror can be used.

```
$ db2inidb XIV as mirror
DBT1000I  The tool completed successfully.
```

7. Roll forward the database to the end of the logs and check whether a database connect works (Example 2-27).

Example 2-27 Database roll forward and check

```
$ db2 rollforward db XIV complete
$ db2 connect to XIV

Database Connection Information
 Database server        = DB2/AIX64 9.7.0
 SQL authorization ID   = DB2XIV
 Local database alias   = XIV
```

3

Volume copy

One of the powerful software features included with the XIV Storage System is volume copy. A volume copy is different from a snapshot because a snapshot creates a point in time copy that is a child device of the original source volume, whereas a volume copy is a point in time copy that is independent of its source volume. This effectively makes it similar to a traditional volume copy, but combined with the sophisticated use of metadata that is found in XIV.

Volume copy's main advantage over a snapshot is that it is independent and would not be at risk of being automatically deleted should pool space become constrained. A volume copy target can also be in a different pool than the source. However, for temporary copies of data with low change rates, a volume copy will most likely be less capacity efficient than using the XIV snapshot function. This is because it effectively duplicates all the data from the source volume at the time it is created.

© Copyright IBM Corp. 2014. All rights reserved.

3.1 Volume copy architecture

The volume copy feature provides an instantaneous copy of data from one volume to another volume. By utilizing the same functionality of the snapshot, the system modifies the target volume to point to the source volume's data. After the pointers are modified, the host has full access to the data on the volume. This means a volume can start using a volume copy immediately.

After the XIV Storage System completes the setup of the pointers to the source data, a background copy of the data is performed. The data is copied from the source volume to a new area on the disk, and the pointers of the target volume are then updated to use this new space. The copy operation is done in such a way as to minimize the impact to the system. If the host performs an update before the background copy is complete, a redirect on write occurs, which allows the volume to be readable and writable before the volume copy completes.

3.2 Performing a volume copy

Performing a volume copy is a simple task. The only requirements are that the target volume must be created and formatted before the copy can occur. This differs from a snapshot, where the target does not exist before snapshot creation.

A nice feature is that the target volume does not initially have to be the same size as the source volume. It can initially be either smaller or larger. The XIV will automatically re-size the target volume to be the same size as the source volume without prompting the user. This presumes there is sufficient space in the pool containing the resized target volume; if there is insufficient space in a particular pool to re-size potential target volumes, then volumes from that pool will be displayed in the GUI pop-up.

Figure 3-1 on page 47 illustrates how you can create a copy of volume `ITSO_Vol1`. The target volume for this example will be `ITSO_Vol2`. First, right-click the source volume to open the command-list menu. From the list, select **Copy this Volume**. A dialog box opens that lists all potential target volume candidates.

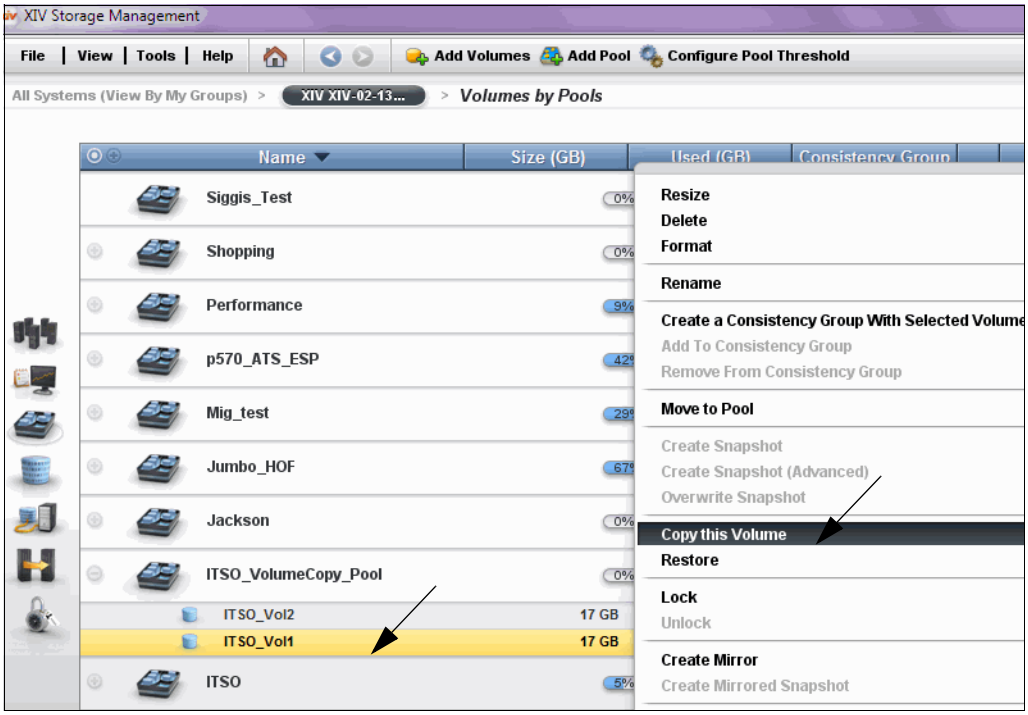


Figure 3-1 Initiating a volume copy process

From the dialog box, select a destination or target volume. Figure 3-2 shows that `ITSO_Vol2` is selected. After selecting your volume, click **OK**. The system prompts you to validate the copy action. The XIV Storage System instantly performs the copy process (using a process known as a metadata update) and displays a command execution completion message. After the completion message is received, the volume is available for use, in that it can be mapped to a host, and then read and write I/O can be directed to it.

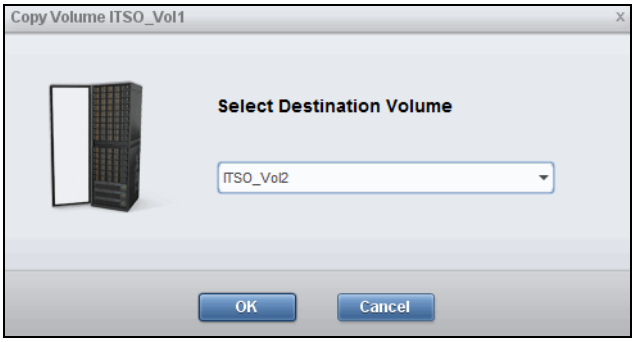


Figure 3-2 Target volume selection

To create a volume copy using the XCLI, the source and target volumes must be specified in the command. If you are running the volume copy command in a script then the **-y** parameter must be specified because the command runs interactively. The **-y** parameter will suppress the "*are you sure*" validation question. For typical script syntax, see Example 3-1.

Example 3-1 Performing a volume copy

```
xcli -c "XIV LAB 01 EBC"-y vol_copy vol_src="ITSO_Vol1" vol_trg="ITSO_Vol2"
```

3.2.1 Monitoring the progress of a volume copy

With XIV volume copy, a background copy occurs after the copy command is processed. There is no way in the XIV GUI or XCLI to display the progress of the background copy, nor is there any need to. The target volume is immediately available for use, whereas a background process will, over the course of time, detect what blocks need to be copied and then duplicate them.

3.3 Troubleshooting issues with volume copy

If the intended target volume is not displayed in the GUI pop-up menu then you might have one of the following issues:

- The target volume has snapshots. Because the target will be overwritten by the source, snapshots of the target will be rendered useless. Remove the snapshots and attempt to create the copy again.
- The target is smaller than the source, but there is not enough space in the pool to resize the target. Increase the size of the pool and attempt to create the copy again.

3.3.1 Using previously used volumes

You might get an error message that the target volume is not formatted, as shown in Figure 3-3. This occurs when there is already data on the target volume.

Figure 3-3 Volume is not formatted error message

This error might occur because of the following possible reasons:

- A previous volume copy might have been performed onto that target volume.
- The target volume has been, or still is, mapped to a host and there is actual host data on the volume.

To work around this error you need to format the volume. Do this by selecting **Volumes** → **Volumes and Snapshots**, right-click the volume to select it and then choose the option to format the volume, as shown in Figure 3-4. You will get a warning prompt that this might cause data loss.

Important: Formatting a volume with data on it will delete that data. Do not format the volume unless you are certain the data on that volume is no longer required.

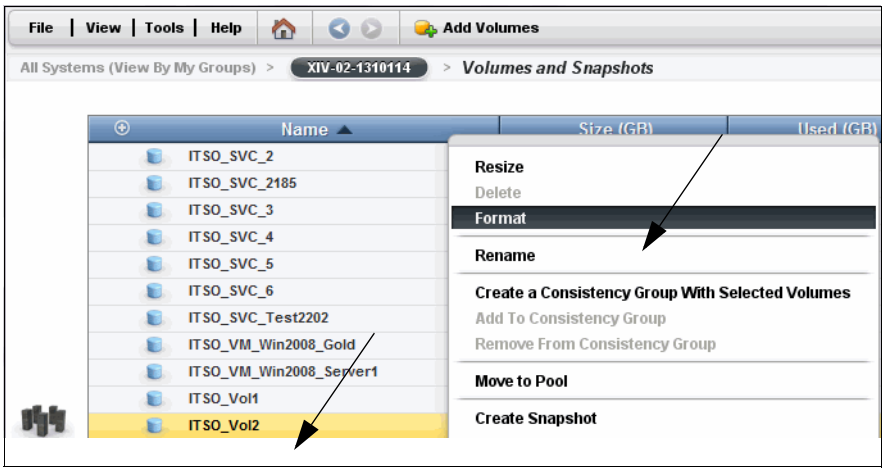


Figure 3-4 Formatting a volume to allow volume copy to occur

3.4 Cloning boot volumes with XIV volume copy

This section describes a possible use case for the XIV volume copy feature. If you create a boot from SAN volume that you consider your *gold copy* (one that is to be used as a basis for deployment) then you might want to deploy it to other servers using volume copy. By using volume copy, the additional server instances can be provisioned without waiting for the operating system to be installed onto each boot disk. In the following examples VMware as the hypervisor is shown. However, this example can be applied to any operating system (OS) installation in which the hardware configuration is similar.

VMware allows the resources of a server to be separated into logical virtual systems, each containing its own OS and resources. In this example each VMware virtual machine will boot from its own SAN boot disk, resident on XIV. The boot disks will be mapped using VMware Raw Device Mapping (RDM), which is also labeled in the vSphere client as a mapped raw LUN.

Figure 3-5 shows a new virtual machine, Win2008_Gold, was created with a SAN boot LUN that is a mapped raw LUN from the XIV. For this example, Windows 2008 was installed onto that disk.

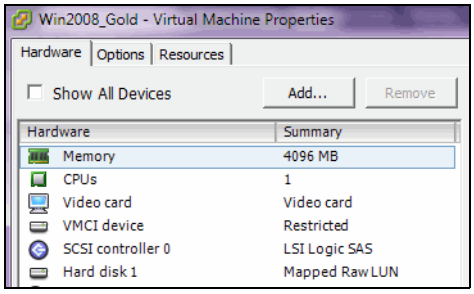


Figure 3-5 Configuration of the virtual machine in VMware

To perform the volume copy, complete the following steps:

1. Validate the configuration for your host. With VMware, ensure that the hard disk assigned to the virtual machine is a mapped raw LUN. For a disk directly attached to a server, the SAN boot must be enabled and the target server must have the XIV volume discovered.
2. Shut down the source server or OS. If the source OS remains active, there might be data in memory that is not synchronized to the disk. If this step is skipped, unexpected results can occur.
3. Perform volume copy from the source volume to the target volume.
4. Map the volume copy to the new system and perform system boot.

A demonstration of the process is simple using VMware. Starting with the VMware resource window, power off the virtual machines for both the source and the target (given that the target is getting a new boot disk, it might not be powered on). The summary in Figure 3-6 shows the guest with the source volume (Win2008_Gold) and the guest with the target volume (Win2008_Server1) are both powered off.

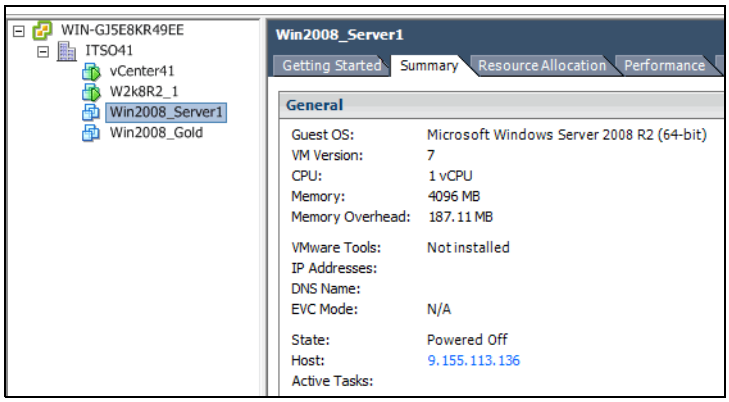


Figure 3-6 VMware virtual machine summary

By looking at the XIV Storage System before the copy operation (Figure 3-7 on page 51), you see that ITSO_VM_Win2008_Gold, which is mapped to the vSphere server and then allocated by vSphere to the Win2008_Gold virtual machine in VMware, used 7 GB of space. This information suggests that the OS is installed.

The second volume, ITSO_VM_Win2008_Server1, is the target volume for the copy. It is mapped to the vSphere server and then allocated by vSphere to the Win2008_Server1

Virtual Machine, and is 0 bytes in size, indicating that at this point the OS has not been installed on the virtual machine and thus the Win2008_Server1 virtual machine is not usable.

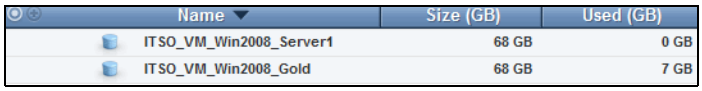


Figure 3-7 The XIV volumes before the copy

Because the virtual machines are powered off, initiate the copy process by selecting ITSO_VM_Win2008_Gold as the source volume to be copied to the ITSO_VM_Win2008_Server1 target volume. The copy completes immediately and the ITSO_VM_Win2008_Server1 target volume is now available for use. One way to verify that the copy command completed is to note that the used area of the volumes now match, as shown in Figure 3-8 where the used value is 7 GB for each volume.

Figure 3-8 The XIV volumes after the copy

Because the copy command is complete, you can power up the new virtual machine to use the newly cloned operating system. Both servers usually boot up normally with only minor modifications to the host. In this example, the server name has to be changed because there are now two servers in the network with the same name. See Figure 3-9.

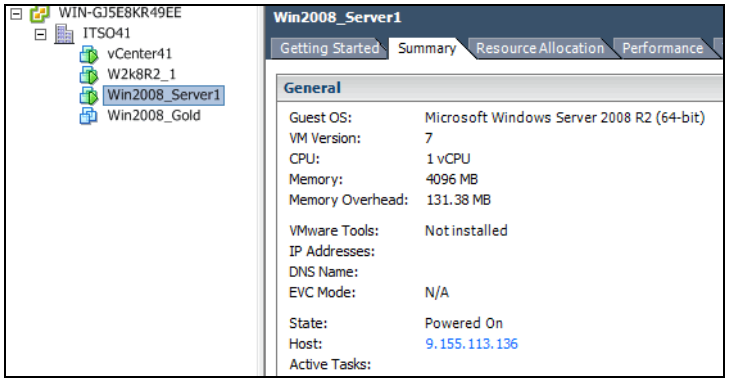


Figure 3-9 VMware summary showing both virtual machines powered on

Figure 3-10 shows the second virtual machine console with the Windows operating system powered on.

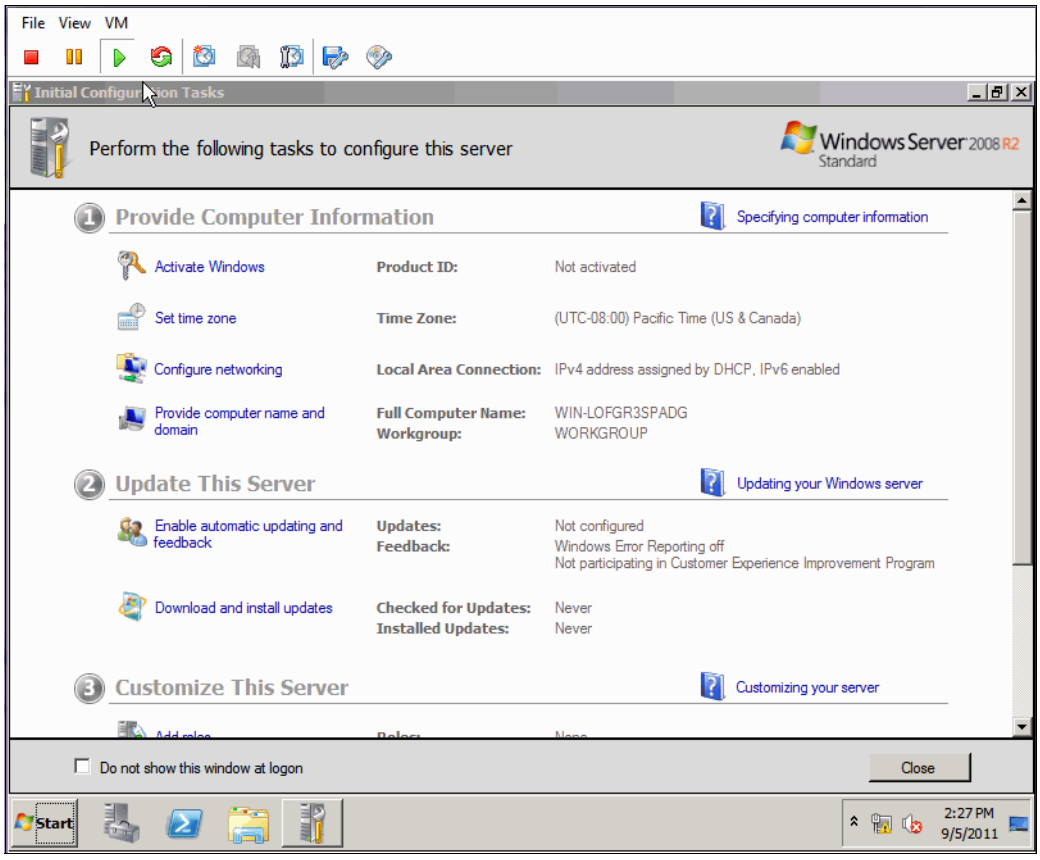


Figure 3-10 Booted Clone Windows server created using volume copy

4

Remote mirroring

The remote mirroring function of the XIV Storage System provides a real-time copy between two or more XIV storage systems supported over Fibre Channel (FC) or iSCSI links. This feature provides a method to protect data from site failures.

Remote mirroring can be a synchronous copy solution where a write operation is completed on both copies (local and remote sites) before an acknowledgement is returned to the host that issued the write (see Chapter 5, “Synchronous Remote Mirroring” on page 113). This type of remote mirroring is typically used for geographically close sites, to minimize the effect of I/O delays that are proportional to the distance between them.

Remote mirroring can also be an asynchronous solution where consistent sets of data are copied to the remote location at predefined intervals while the host I/O operations are acknowledged directly after they are written on the primary site alone (see Chapter 6, “Asynchronous remote mirroring” on page 149). This is typically used for longer distances between sites.

Important: For mirroring, a reliable, dedicated network is preferred. Links can be shared, but require available and consistent network bandwidth. The specified minimum bandwidth (10 Mbps for FC and 50 Mbps for iSCSI as per XIV release 11.1.x) is a functional minimum and does not necessarily meet the replication speed required for a given customer environment and workload.

Also, minimum bandwidths are *not* time-averaged, as typically reported by network monitoring packages, but are instantaneous, constant requirements, typically achievable only through network quality of service (QoS) or similar.

Unless otherwise noted, this chapter describes the basic concepts, functions, and terms that are common to both XIV synchronous and asynchronous mirroring.

© Copyright IBM Corp. 2014. All rights reserved.

4.1 XIV Remote mirroring overview

The purpose of mirroring is to create a set of consistent data that can be used in the event of problems with the production volumes, or for other purposes such as testing and backup on the remote site using snapshots of a consistent data.

XIV remote mirroring is independent of application and operating system, and does not require server-processor cycle usage.

Note: Although not depicted in most of the diagrams, a switch is required to connect the XIV storage systems being mirrored, which means a direct connection is not supported.

4.1.1 XIV remote mirror terminology

To become familiar with the mirroring related terms used in this book, their definitions are outlined in the following list:

- *Local site:* This site consists of the primary XIV storage and the servers running applications stored on that XIV system.
- *Remote site:* This site holds the mirror copy of the data using XIV Storage System and usually also has standby servers. A remote site is capable of becoming the active production site using a consistent data copy.
- *Primary*: This denotes the XIV used for production, during typical business routines, to serve hosts and have its data replicated to a secondary XIV,.
- *Secondary*. This denotes the XIV used during normal circumstances to act as the mirror (backup) for the primary.
- *Consistency group (CG)*: This is a set of related volumes on a single system that are treated as one logical unit. Thus all CG's data reflects correctly ordered writes across all respective volumes within the CGs. Consistency groups are supported within remote mirroring.
- *Coupling*: This is the *pairing* of volumes or CGs to form a mirror relationship between the source of a replication (master) and its target (slave).
- *Peer*: This is one side of a coupling. It can be either a volume or a consistency group. However, peers must be of the same type (that is, both volumes or CGs). Whenever a coupling is defined, a role is specified for each peer. One peer is designated as the master and the other peer is designated as the slave.
- *Role*: This denotes the actual role that the peer is fulfilling:
  - *Master*: A role that indicates that the peer serves host requests and acts as the source for replication. Changing a peer's role to master from slave might be warranted after a disruption of the current master's service, either because of a disaster or a planned service maintenance.
  - *Slave*: A role that indicates that the peer does not serve host write requests (it can used in read-only mode) and acts as the target for replication. Changing a peer's role to slave from master might be warranted after the peer is recovered from a site, system, or link failure, or disruption that led to the promotion of the other peer from slave to master. Changing roles can also be done in preparation for supporting a planned service maintenance.

- *Sync job*: This applies to asynchronous mirroring only. It denotes a synchronization procedure run by the master at user-configured intervals corresponding to the asynchronous mirroring definition, or upon manual execution of the XCLI command **`mirror_create_snapshot`** (which is used also for synchronous mirroring but not as part of a scheduled job). The resulting job is referred to as *snapshot mirror sync job* or *ad-hoc* sync job, or *manual sync job* in contrast with a scheduled sync job. The sync job entails synchronization of data updates recorded on the master since the creation time of the most-recent snapshot that was successfully synchronized.
- *Offline initialization (offline init)*: A mechanism whereby XIV through the use of HASH values compares respective source and target 64 KB data blocks and copies over only the chunks that have different data. Offline init aims at expediting the initialization of mirror pairs that are known to be inherently similar (for example, when a synchronous mirror pair is changed to an asynchronous pair, or asynchronous pair is changed to synchronous pair). This unique feature of XIV is evident when the data links do not have adequate speed or capacity to transmit the entire volume in a timely fashion. In that case, the pair is first created when the machines are at close proximity and can use fast links. Then, when the XIV machine that hosts the remote mirror is placed at its final physical destination, only the changed data since those volumes were identical, has to be copied over the wire. Offline initialization or *trucking* is described in “Offline initialization” on page 156.
- *Asynchronous schedule interval*: This applies only to asynchronous mirroring. It represents, per a coupling, how often the master automatically runs a new sync job. The default interval and the minimum possible is 20 seconds.
- *Recovery point objective (RPO)*: The RPO is a setting that is applicable only to asynchronous mirroring. It represents an objective set by the user, implying the maximal currency difference considered acceptable between the mirror peers (the actual difference between mirror peers can be shorter or longer than the RPO set).

  An RPO of zero indicates that no difference between the mirror peers can be tolerated and that implies that sync mirroring is required. An RPO that is greater than zero indicates that the replicated volume is less current or lags somewhat behind the master volume, and that there is a potential for certain transactions that have been run against the production volume to be rerun when applications start to use the replicated volume.

  For XIV asynchronous mirroring, the required RPO is user-specified. The XIV system then reports effective RPO and compares it to the required RPO.

  Connectivity, bandwidth, and distance between the XIV systems directly impact RPO. More connectivity, greater bandwidth, and less distance typically enable a lower RPO.

4.1.2 XIV remote mirroring modes

As mentioned in the introduction, XIV supports both synchronous mirroring and asynchronous mirroring:

- XIV synchronous mirroring

  XIV synchronous mirroring is designed to accommodate a requirement for zero RPO.

  To ensure that data is also written to the secondary XIV (slave role), an acknowledgement of the write operation to the host is issued only after the data has been written to both XIV systems. This ensures the sameness of mirroring peers at all times. A write acknowledgement is returned to the host, only after the write data has been cached by two separate XIV modules at each site. This is depicted in Figure 4-1.

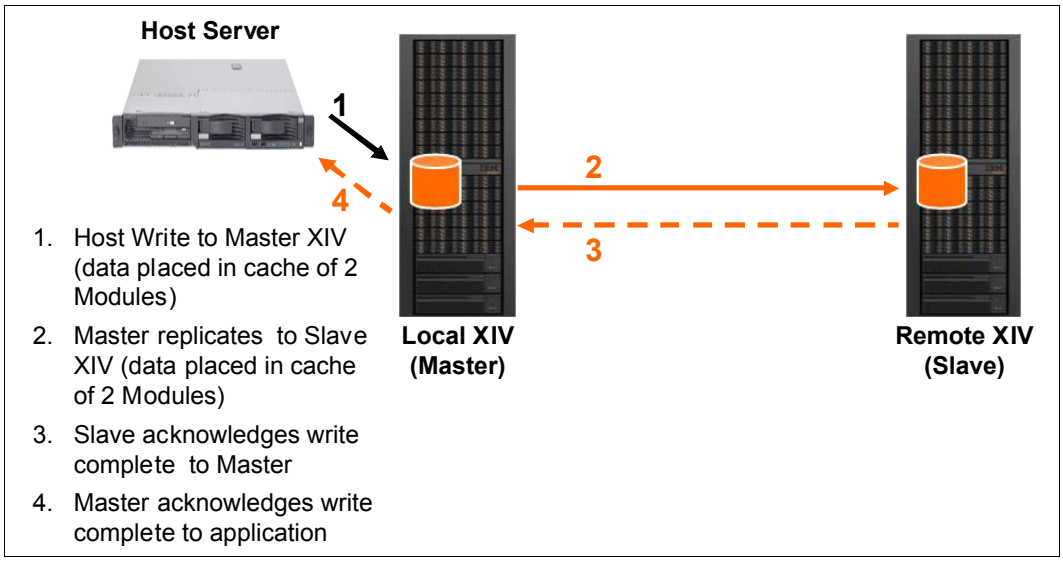

Host Server
1
2
3
4
1. Host Write to Master XIV (data placed in cache of 2 Modules)
2. Master replicates to Slave XIV (data placed in cache of 2 Modules)
3. Slave acknowledges write complete to Master
4. Master acknowledges write complete to application
Local XIV (Master)
Remote XIV (Slave)

Figure 4-1 XIV synchronous mirroring

Host read operations are provisioned by the primary XIV (master role), whereas writing is performed at the primary (master role) and replicated to the secondary XIV systems. See 5.8.1, “Disaster recovery scenario with synchronous mirroring” on page 135, for more details.

- XIV asynchronous mirroring

  XIV asynchronous mirroring is designed to provide a consistent replica of data on a target peer through timely replication of data changes recorded on a source peer.

  XIV Asynchronous mirroring exploits the XIV snapshot function, which creates a point-in-time (PiT) image. In XIV asynchronous mirroring, successive snapshots (point-in-time images) are made and used to create consistent data on the slave peers. The system sync job copies the data corresponding to the differences between two designated snapshots on the master (most-recent and last-replicated).

For XIV asynchronous mirroring, acknowledgement of *write complete* is returned to the application as soon as the write data has been received at the local XIV system, as shown in Figure 4-2. See 6.6, “Detailed asynchronous mirroring process” on page 176, for details.

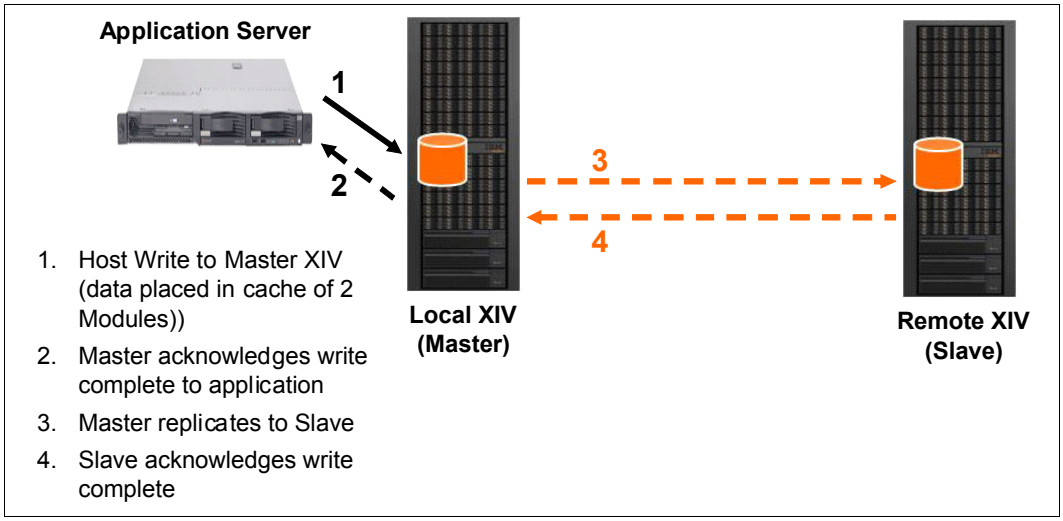


Figure 4-2 XIV asynchronous mirroring

4.2 Mirroring schemes

Mirroring, whether synchronous or asynchronous, requires two or more XIV systems. The source and target of the asynchronous mirroring can reside at the same site and form a local mirroring or they can reside at different sites and facilitate a disaster recovery plan. Figure 4-3 shows how peers can be spread across multiple storage systems and sites.

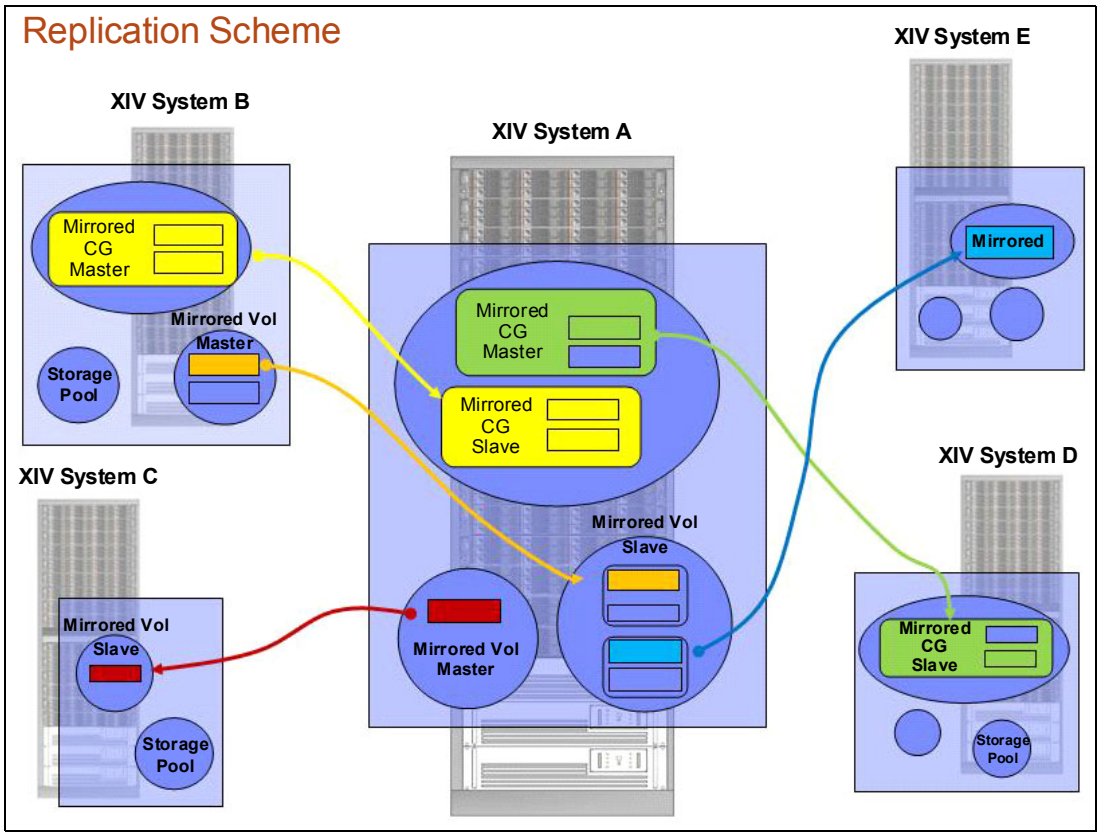


Figure 4-3 Mirroring replication schemes

An XIV system can be connected to up to 8 other XIV targets, for mirroring purposes. An XIV system can be used simultaneously as replication sources and replication targets.

In a bidirectional configuration, an XIV system concurrently functions as the replication source (master) for one or more couplings, and as the replication target (slave) for other couplings. Figure 4-3 illustrates possible schemes for how mirroring can be configured.

Figure 4-4 shows connectivity among systems and group of systems, as shown in the XIV Hyper-Scale Manager GUI.

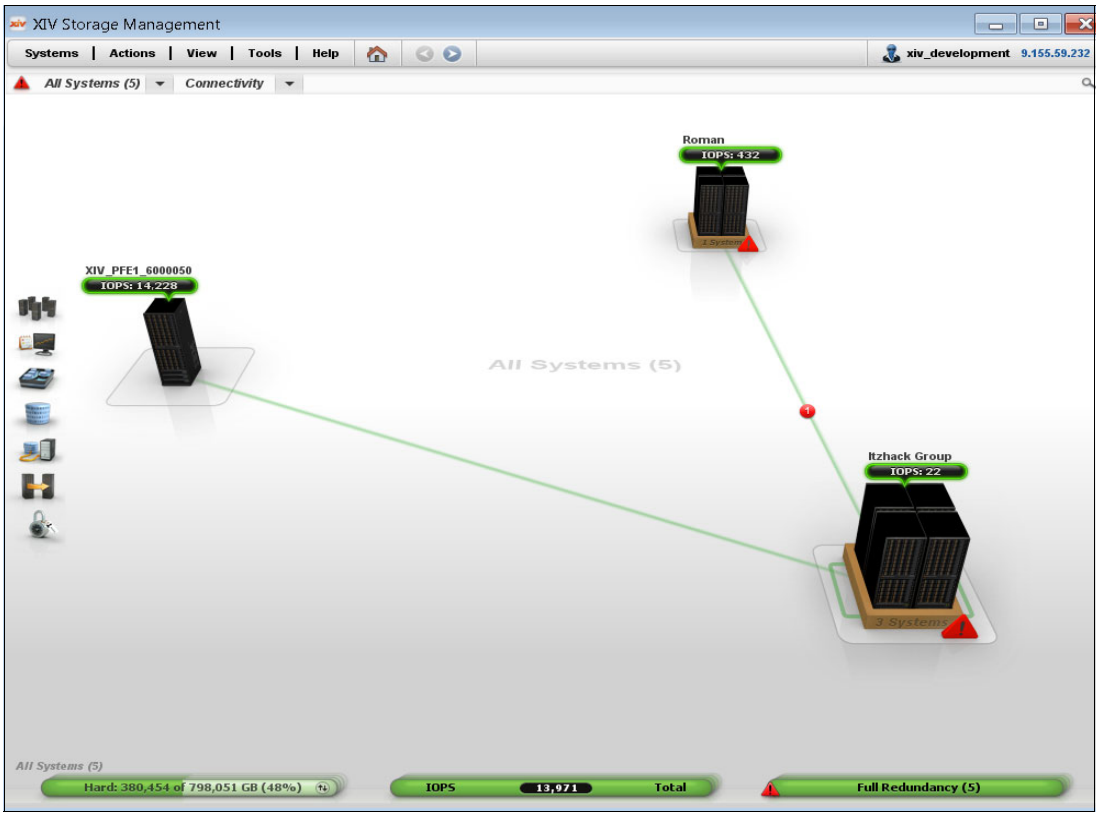


Figure 4-4 XIV GUI connectivity among systems and group of systems in Hyper-Scale Manager

4.2.1 Peer designations and roles

A peer (volume or consistency group) is assigned either a master or a slave role when the mirror is defined. By default, in a new mirror definition, the location of the master denotes the primary system, and the slave denotes the secondary system. An active mirror must have exactly one primary and exactly one secondary.

Important: A single XIV can contain both master volumes and CGs (mirroring to another XIV) and slave volumes and CGs (mirroring from another XIV). Peers in a master role and peers in a slave role on the same XIV system must belong to different mirror couplings.

The various mirroring role status options are as follows:

- Designations:
  - Primary: The designation of the source peer, which is initially assigned the master role
  - Secondary: The designation of the target peer, which initially plays the slave role
- Role status:
  - Master: Denotes the peer with the source data in a mirror coupling. Such peers serve host requests and are the source for synchronization updates to the slave peer. Slave and master roles can be *switched* by using the **`mirror_switch_roles`** command, if the status is *synchronized* for synchronous mirror and, it is in an RPO_OK state for an

asynchronous mirror. For both synchronous and asynchronous mirroring, the master can be *changed* (**`mirror_change_role`** command) to a slave if the status is *inactive*.

– Slave: Denotes the target peer in a mirror. Such peers do not serve host write requests and accept synchronization updates from a corresponding master. A slave LUN can be accessed in read-only mode by a host.

Consistency group within an XIV

With mirroring (synchronous or asynchronous), the major reason for consistency groups is to handle a large number of mirror pairs as a *group* (mirrored volumes are consistent). Instead of dealing with many mirror pairs individually, consistency groups simplify the handling of those related pairs considerably.

Important: If your mirrored volumes are in a mirrored consistency group, you cannot do mirroring operations like *deactivate* or *change_role* on a single volume basis. If you want to do this, you must remove the volume from the consistency group; see 5.4, “Mirrored snapshots (ad-hoc sync jobs)” on page 126 or “Removing a volume from a mirrored consistency group” on page 161.

Consistency groups also play an important role in the recovery process. If mirroring was suspended (for example, because of complete link failure), data on different slave volumes at the remote XIV are consistent. However, when the links are up again and resynchronization process is started, data spread across several slave volumes is not consistent until the master state has reached the *synchronized* state. To preserve the consistent state of the slave volumes, the XIV system automatically creates a snapshot of each slave volume and keeps it until the remote mirror volume pair is synchronized (the snapshot is kept until all pairs are synchronized in order to enable restoration to the same consistent point in time). If the remote mirror pairs are in a consistency group, then the snapshot is taken for the whole group of slave volumes and the snapshots are preserved until *all* pairs are synchronized. Then, the snapshot is deleted automatically.

4.2.2 Operational procedures

Mirroring operations involve configuration, initialization, ongoing operation, handling of communication failures, and role switching activities.

The following list defines the mirroring operation activities:

- Configuration

  Local and remote replication peers are defined by an administrator who specifies the master and slave peers roles. These peers can be volumes or consistency groups. The secondary peer provides a backup of the primary.

- Initialization

  Mirroring operations begin with a master volume that contains data and a formatted slave volume. The first step is to copy the data from the master volume (or CG) to the slave volume (or CG). This process is called *initialization*. Initialization is performed once in the lifetime of a mirror. After it is performed, both volumes or CGs are considered to be synchronized to a specific point in time. The completion of initialization marks the first point-in-time that a consistent master replica on the slave is available. Details of the process differ depending on the mirroring mode (synchronous or asynchronous). See 5.8.1, “Disaster recovery scenario with synchronous mirroring” on page 135 (for synchronous mirroring), and 6.6, “Detailed asynchronous mirroring process” on page 176 (for asynchronous mirroring).

- Offline initialization

  Offline initialization operation begins with a master volume that contains data and a slave volume, which also contains data and is related to this same master. At this step, only different chunks are copied from the master to its slave. Offline initialization can be performed whenever mirror pair was suspended or when the mirror type changes from synchronous to asynchronous and from asynchronous to synchronous.

- Mirror mode switching

  Before version 11.4, it was possible to leverage the offline initialization for switching between a synchronous mirror to an asynchronous one. Starting with version 11.4, the offline initialization can be used for both directions.The toggling between the two modes implies the deactivation of the incumbent mirror mode and, the deletion of the mirror pair and also of the respective snapshots on both ends and, unlocking of the slave mirror. Then, the new mode is selected and a new is created between the peers. Using the offline initialization, only the new data that was written to the primary XIV, since the deletion of the original mirror, is copied over. Thus, the toggling between the two operational modes does not require a redundant full copy, which was the case before when switching between asynchronous mirror mode to a synchronous one.

- Ongoing operation

  After the initialization process is complete, mirroring ensues.

  In synchronous mirroring, normal ongoing operation means that all data written to the primary volume or CG is first mirrored to the slave volume or CG. At any point in time, the master and slave volumes or CGs will be identical except for any unacknowledged (pending) writes.

  In asynchronous mirroring, ongoing operation means that data is written to the master volume or CG and, is replicated to the slave volume or CG at specified intervals.

- Monitoring

  The XIV System effectively monitors the mirror activity and places events in the event log for error conditions. Alerts can be set up to notify the administrator of such conditions. You must have set up SNMP trap monitoring tools or email notification to be informed about abnormal mirroring situations.

- Handling of communication failures

  Sometimes the communication between the sites might break down. The master continues to serve host requests as the synchronous mirroring is based on best effort, in attempt to minimize the impact to the hosts. Upon recovery from a link down incident, the changed data is copied over and, mirroring is resumed. Events are generated for link failures.

- Role switching

  If required, mirror peer roles of the slave and master can be switched. Role switching is always initiated at the master site. Usually, this is done for certain maintenance operations or because of a test drill that verifies the disaster recovery (DR) procedures. Role switching should be used cautiously especially with asynchronous mirroring. When roles are switched for an asynchronous mirror, data can be lost for an interval up to the RPO time because the remote site is typically lagging, in time, for a given asynchronous pair. Role switching in the case of synchronous mirror is designed so that no data loss can occur. Role switching should be used only for cases such as a catastrophic host failure at the master site when the pairing is intact, but there have been no write operations to the master since the last sync job was completed.

- Role changing

  In case of a disaster at the primary site, the master peer might fail. To allow read/write access to the volumes at the remote site, the volume's role must be changed from slave to master. A role *change* changes only the role of the XIV volumes or CGs to which the command was addressed. Remote mirror peer volumes or CGs are not changed automatically. That is why changing roles on both mirror sides if mirroring is to be restored is imperative (if possible).

4.2.3 Mirroring status

The status of a mirror is affected by several factors, such as the links between the XIVs or the initialization state.

Link status

The link status reflects the connection from the master to the slave volume or CG. A link has a direction (from local site to remote or vice versa). A failed link or a failed secondary system both result in a link error status. The link state is one of the factors determining the mirror operational status. Link states are as follows:

- OK: Link is up and is functioning.
- Error: Link is down.

Figure 4-5 and Figure 4-6 depict how the link status is reflected in the XIV GUI, respectively.

Figure 4-5 Link up

Figure 4-6 Link down

If several links (at least two) are in one direction and one link fails, this usually does not affect mirroring if the bandwidth of the remaining link is high enough to keep up with the data traffic.

Monitoring the link utilization

The mirroring bandwidth of the links must be high enough to cope with the data traffic generated by the changes on the master volumes. During the planning phase, before setting up mirroring, monitor the write activity to the local volumes. The bandwidth of the links for mirroring must be as large as the peak write workload.

After mirroring has been implemented, from time to time monitor the utilization of the links. The XIV statistics panels allow you to select *targets* to show the data traffic to remote XIV systems, as shown in Figure 4-7.

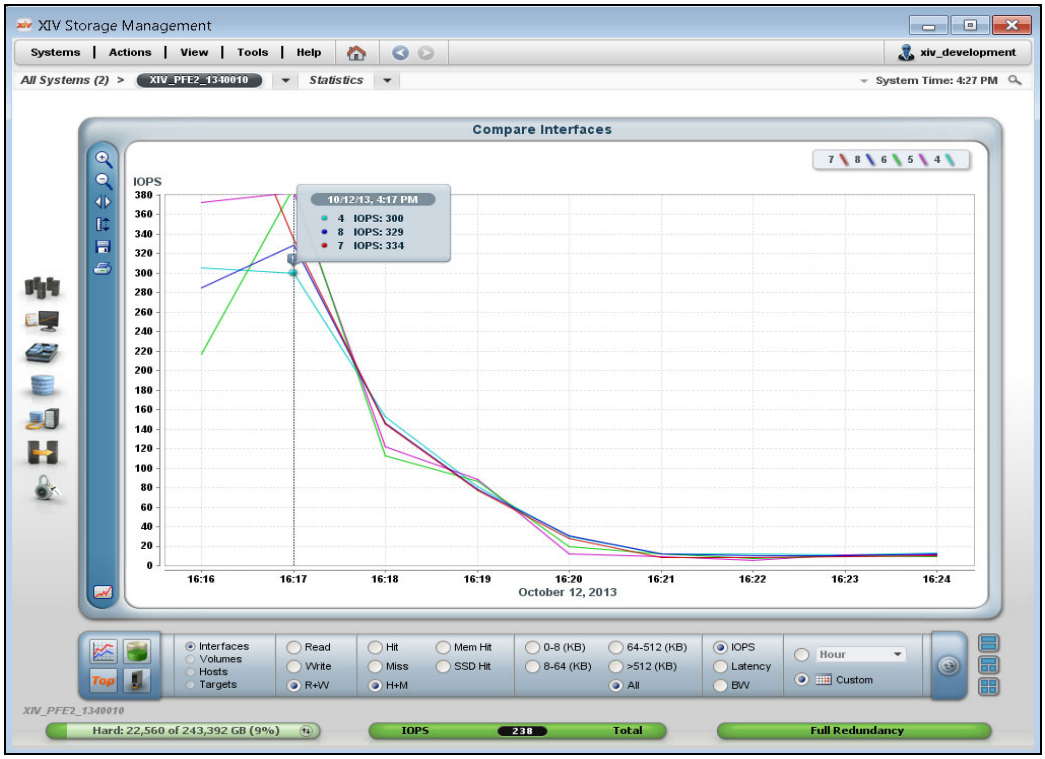


Figure 4-7 Monitoring link utilization, with pop-up flyover shown

Mirror operational status

Mirror operational status is defined as either *operational* or *non_operational*.

- Mirroring is operational in the following situations:
  - The activation state is *active*.
  - The link is UP.
  - Both peers have different roles (master or slave).
  - The mirror is active.
- Mirroring is non_operational in the following situations:
  - The mirror is inactive.
  - The link is in an error state or deactivated (link down).

Synchronous mirroring states

Note: This section applies only to synchronous mirroring.

The synchronization status reflects whether the data of the slave's volume is identical to the master's volume. Because the purpose of the remote mirroring feature is to ensure that the slave's volume is an exact copy of the master's volume, this status indicates whether this objective is currently being achieved.

The following states or statuses are possible:

- Initializing

  The first step in remote mirroring is to create a copy of all the data from the master volume or CG to the slave volume or CG. During this initial copy phase, the status remains *initializing*.

- Synchronized (master volume or CG only)/consistent (slave volume or CG only)

  This status indicates that all data that has been written to the master volume or CG has also been written to the slave volume or CG. Ideally, the master and slave volumes or CGs must always be synchronized. However, this does not always indicate that the two volumes are absolutely identical in case of a disaster because there are situations when there might be a limited amount of data that was written to one volume, but that was not yet written to its peer volume. This means that the write operations have not yet been acknowledged to the respective hosts. Such writes are known as *pending writes* or *data in flight*.

- Unsynchronized (master volume only)/inconsistent (slave volume only)

  After a volume or CG has completed the initializing stage and achieved the synchronized status it can become *unsynchronized* (master) or *inconsistent* (slave). This occurs when it is not known whether all the data that has been written to the master volume has also been written to the slave volume. This status can occur in the following cases:

  – The communications link is down and as a result certain data might have been written to the master volume, but was not yet written to the slave volume.

  – Secondary XIV is down. This is similar to communication link errors because in this state, the Primary XIV is updated, whereas the secondary is not.

  – Remote mirroring is deactivated. As a result, certain data might have been written to the master volume and not to the secondary volume.

The XIV keeps track of the partitions that have been modified on the master volumes and when the link is operational again or the remote mirroring is reactivated, these changed partitions are sent to the remote XIV and applied to the respective slave's volumes.

Asynchronous mirroring states

Note: This section applies only to asynchronous mirroring.

The mirror state can be either inactive or initializing:

- Inactive: The synchronization process is disabled. It is possible to delete a mirror in this state.
- Initializing: The initial copy is not done yet. Synchronization does not start until the initialization completes. The mirror cannot be deleted during this state.

Important: In cases of an unstable data link, it is possible for the initialization to restart. In this case, the progress bar will return to the left side of the display. This does not mean that the initialization is starting again from the beginning. On the restart of a mirror initialization, the initialization resumes where it left off, and the progress bar will display the percent complete of the remaining data to copy, not the percentage of the full volume.

When initialization is complete, the synchronization process is enabled. Then, it is possible to run sync jobs and copy data between master and slave. Synchronization states are as follows:

- RPO_OK: Synchronization completed within the specified sync job interval time (RPO).
- RPO_Lagging: Synchronization completed but took longer that the specified interval time (RPO).

4.3 XIV remote mirroring usage

Remote mirroring solutions can be used to address multiple types of failures and planned outages. The failure scenarios vary. They can be a result of events that affect a single XIV system. The failure can stem from events that affect an entire data center or campus. Worse, they can be caused by events that affect a whole geographical region. One strategy is to be prepared for all three scenarios. To this end, the disaster recovery (DR) XIV systems are located in three sites. One DR system can even be in the same room as the production system is. The second XIV might be in the same vicinity although not in the same building. The third system can be much farther away. This strategy provides layered recovery protection. Figure 4-8 shows such a recovery plan solution that provides protection from these three failure types, single-system failures, local disasters, and regional disasters.

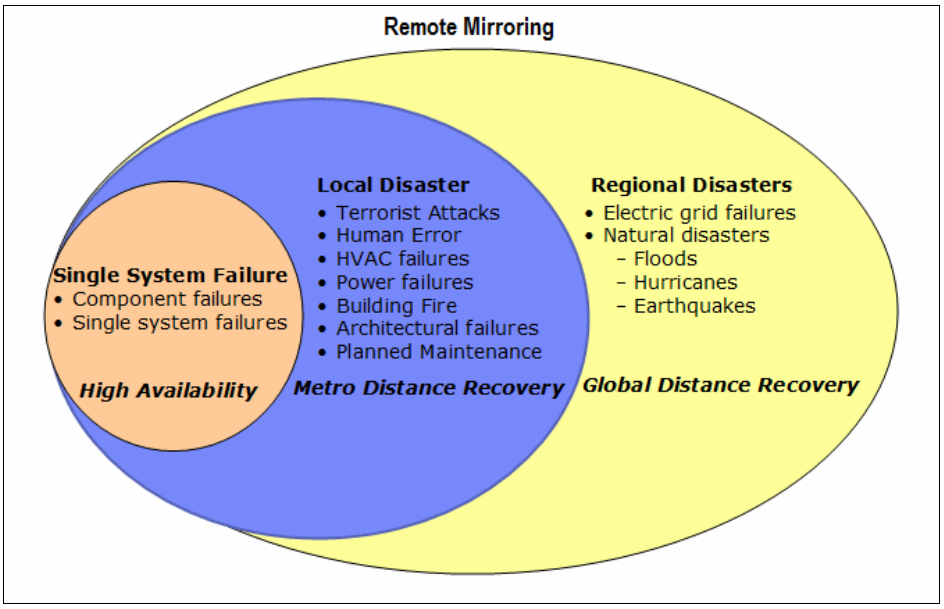


Figure 4-8 Disaster recovery protection levels

Several configurations are possible:

- Single-site high-availability XIV remote mirroring configuration

  Protection for the event of a failure or planned outage of an XIV system (single-system failure) can be provided by a *zero-distance* high-availability (HA) solution including another XIV system in the same location (zero distance). Typical usage of this configuration is an XIV synchronous mirroring solution that is part of a high-availability clustering solution including both servers and XIV storage systems. Figure 4-9 shows a single-site high-availability configuration (where both XIV systems are in the same data center).

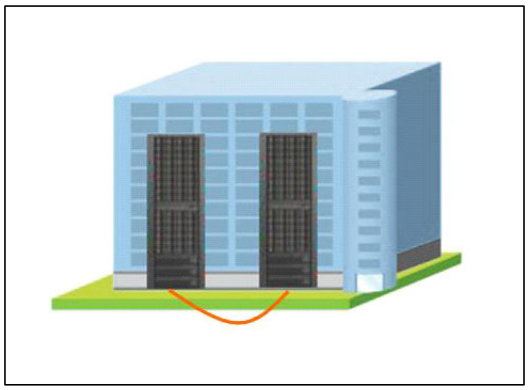


Figure 4-9 Single site HA configuration

- Metro region XIV remote mirroring configuration

  Protection for the event of a failure or planned outage of an entire location (*local disaster*) can be provided by a metro distance disaster recovery solution, including another XIV system in a different location within a metro region. The two XIV systems might be in different buildings on a corporate campus or in different buildings within the same city (typically up to approximately 100 km apart). Typical usage of this configuration is an XIV synchronous mirroring solution. Figure 4-10 shows a metro region disaster recovery configuration.

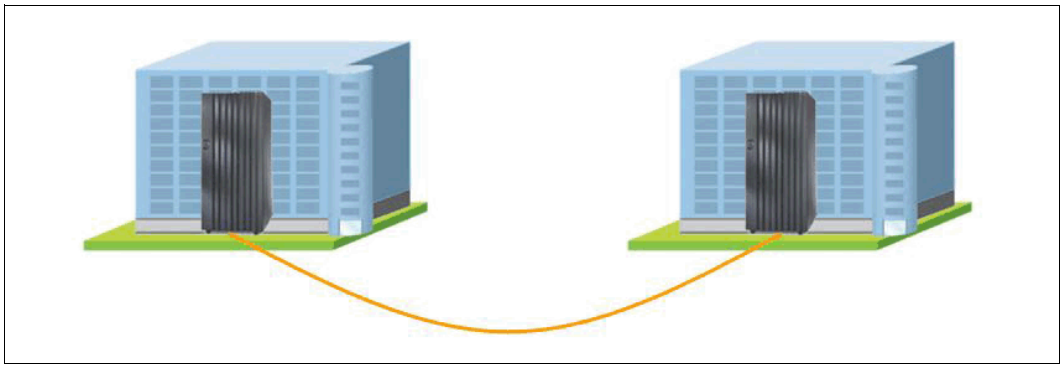


Figure 4-10 Metro region disaster recovery configuration

- Out-of-region XIV remote mirroring configuration

  Protection for the event of a failure or planned outage of an entire geographic region (*regional disaster*) can be provided by a global distance disaster recovery solution including another XIV system in a different location outside the metro region. (The two locations might be separated by up to a global distance.) Typical usage of this configuration is an XIV asynchronous mirroring solution. Figure 4-11 on page 67 shows an out-of-region disaster recovery configuration.

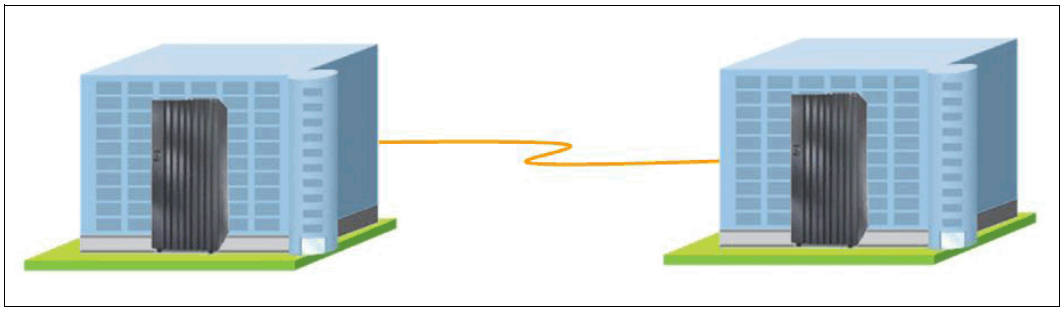


Figure 4-11 Out-of-region disaster recovery configuration

- Metro region plus out-of-region XIV mirroring configuration

  Certain volumes can be protected by a metro distance disaster recovery configuration, and other volumes can be protected by a global distance disaster recovery configuration, as shown in the configuration in Figure 4-12.

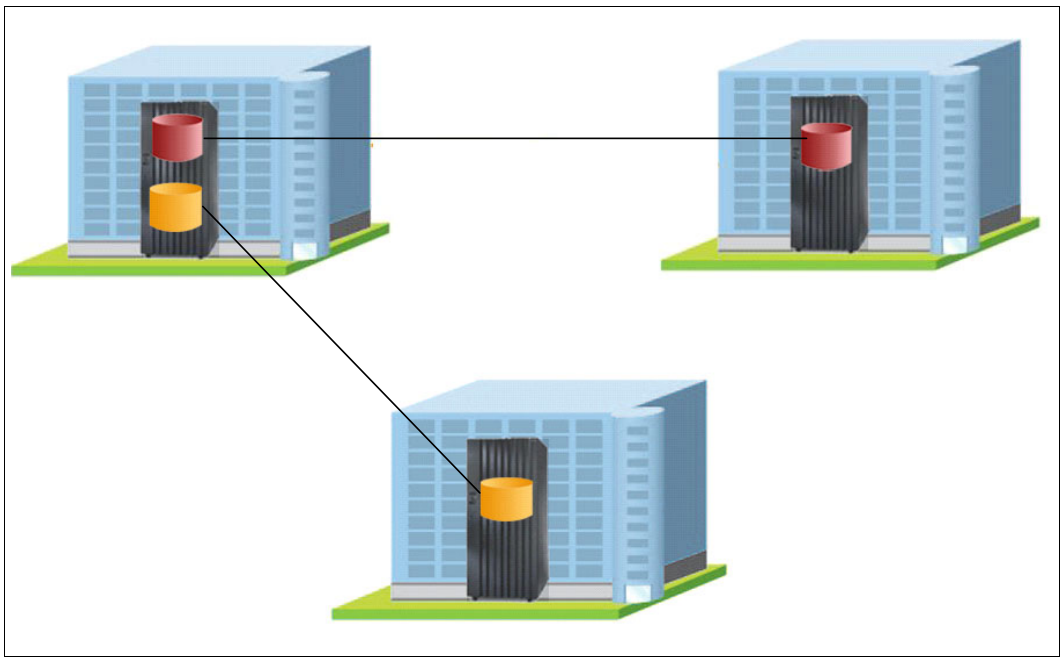


Figure 4-12 Metro region plus out-of-region configuration

Typical usage of this configuration is an XIV synchronous mirroring solution for a set of volumes with a requirement for zero RPO, and an XIV asynchronous mirroring solution for a set of volumes with a requirement for a low, but non-zero RPO. Figure 4-12 shows a metro region for one set of volumes and, another set of volumes using out-of-region configuration.

4.3.1 Using snapshots

Snapshots can be used with remote mirroring to provide copies of production data for business or IT purposes. Moreover, when used with remote mirroring, snapshots provide protection against data corruption. Like any continuous or near-continuous remote mirroring solution, XIV remote mirroring cannot protect against software data corruption because the corrupted data will be copied as part of the remote mirroring solution. However, the XIV snapshot function provides a point-in-time image that can be used for a rapid recovery in the

event of software data corruption (that occurred after the snapshot was taken). The XIV snapshot can be used in combination with XIV remote mirroring, as illustrated in Figure 4-13.

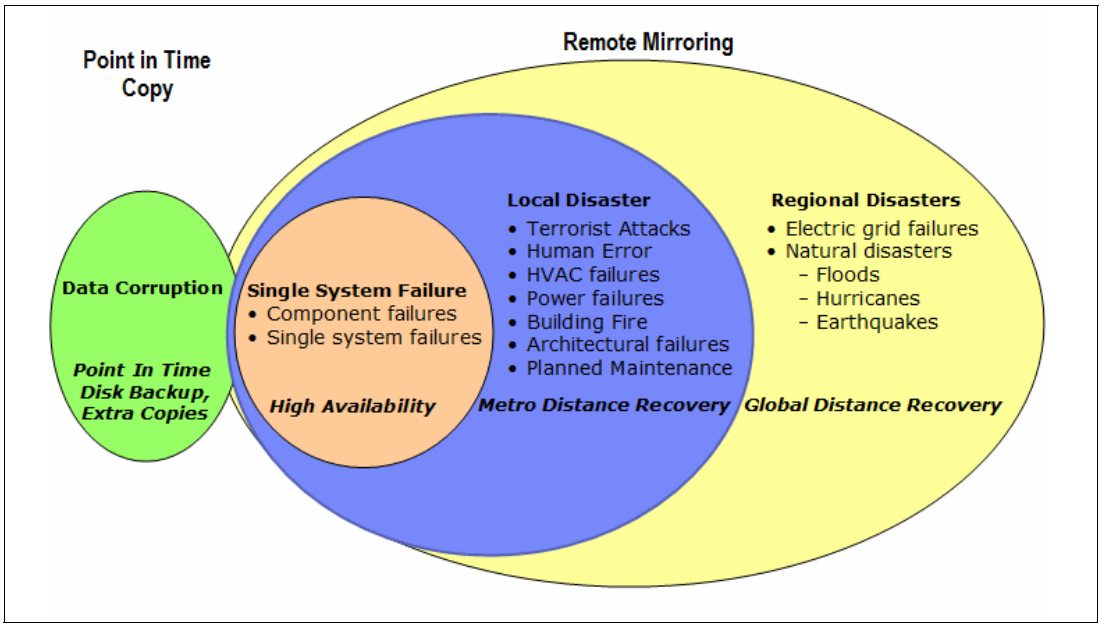


Figure 4-13 Combining snapshots with remote mirroring

Recovery using a snapshot warrants deletion and re-creation of the mirror.

- XIV snapshot (within a single XIV system)

  Protection for the event of software *data corruption* can be provided by restoring the volume to an healthy point-in-time snapshot. The snapshot can be backed up if needed.

- XIV local snapshot plus remote mirroring configuration

  An XIV snapshot of the production (local) volume can be used in addition to XIV remote mirroring of the production volume when protection from logical data corruption is required in addition to protection against failures and disasters. The extra XIV snapshot of the production volume provides a quick restoration to recover from data corruption. An extra snapshot of the production (local) volume can also be used for other business or IT purposes (for example, reporting, data mining, development and test, and so on).

  Figure 4-14 shows an XIV local snapshot plus remote mirroring configuration.

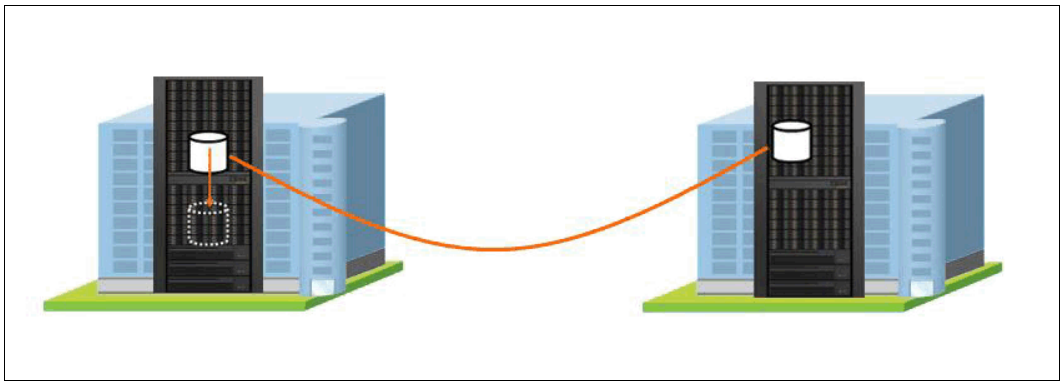


Figure 4-14 Local snapshot plus remote mirroring configuration

- XIV remote snapshot plus remote mirroring configuration

  An XIV snapshot of the consistent replicated data at the remote site can be used in addition to XIV remote mirroring to provide an additional consistent copy of data that can be used for business purposes such as data mining, reporting, and for IT purposes, such as remote backup to tape or development, test, and quality assurance. Figure 4-15 shows an XIV remote snapshot plus remote mirroring configuration.

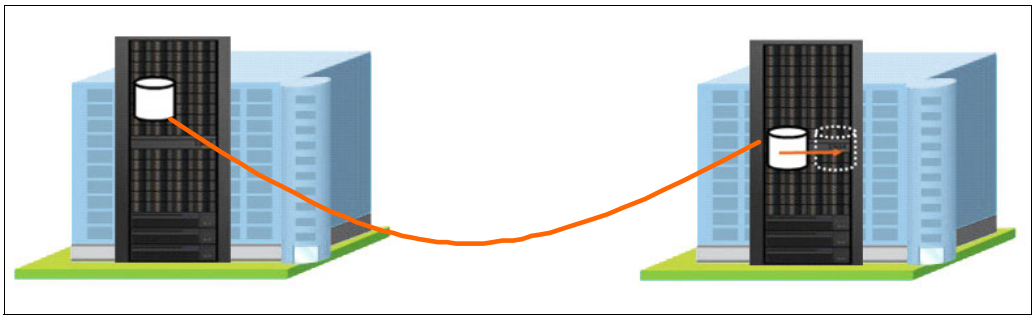


Figure 4-15 XIV remote snapshot plus remote mirroring configuration

4.4 XIV remote mirroring actions

The XIV remote mirroring actions in this section are the fundamental building blocks of XIV remote mirroring solutions and usage scenarios.

4.4.1 Defining the XIV mirroring target

To connect two XIV systems for remote mirroring, each system must be defined to be a mirroring target of the other. An XIV mirroring target is an XIV system with volumes that receive data copied through XIV remote mirroring. Defining an XIV mirroring target for an XIV system simply involves giving the target a name and specifying whether Fibre Channel or iSCSI protocol will be used to copy the data. For a practical illustration, see 4.11.2, “Remote mirror target configuration” on page 104.

XIV remote mirroring copies data from a peer on one XIV system to a peer on another XIV system (the mirroring target system). Whereas the basic underlying mirroring relationship is a one-to-one relationship between two peers, XIV systems can be connected in several ways:

- XIV target configuration: one-to-one

  The most typical XIV remote mirroring configuration is a one-to-one relationship between a local XIV system (production system) and a remote XIV system (DR system), as shown in Figure 4-16. This configuration is typical where there is a single production site and a single disaster recovery (DR) site.

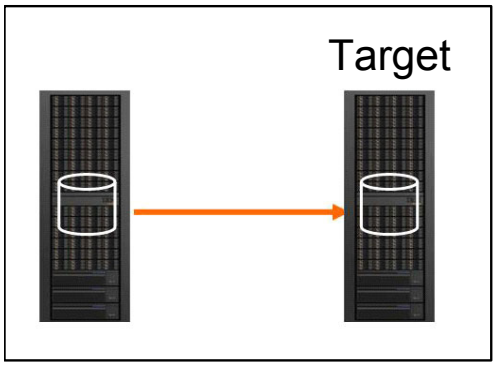


Figure 4-16 One-to-one target configuration

During normal remote mirroring operation, one XIV system (at the DR site) is active as a mirroring target. The other XIV system (at the local production site) will be active as a mirroring target only when it becomes available again after an outage and change of roles between the production and the DR site. Data changes made while production is running on the remote (DR) site, are copied back to the original production site, as shown in Figure 4-17.

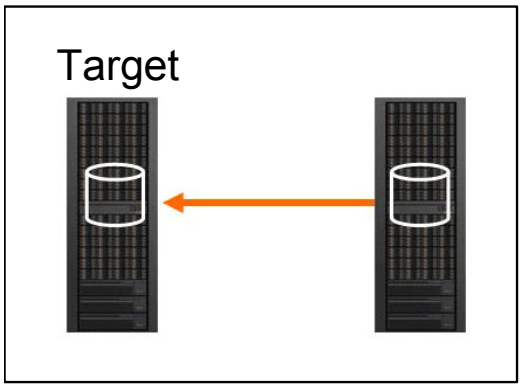


Figure 4-17 Copying changes back to production

In a configuration with two identically provisioned sites, production might be periodically switched from one site to another as part of normal operation, and the XIV system that is the active mirroring target will be switched at the same time. The **`mirror_switch_roles`** command allows for switching roles in both synchronous and asynchronous mirroring. There are special requirements for doing so with asynchronous mirroring.

- XIV target configuration: synchronous and asynchronous one-to-one

  XIV supports both synchronous and asynchronous mirroring (for different peers) on the same XIV system, so a single local XIV system could have certain volumes synchronously mirrored to a remote XIV system, whereas other peers are asynchronously mirrored to the same remote XIV system as shown in Figure 4-18. Highly response-time-sensitive volumes could be asynchronously mirrored and less response-time-sensitive volumes could be synchronously mirrored to a single remote XIV.

Figure 4-18 Synchronous and asynchronous peers

- XIV target configuration: fan-out

  A single local (production) XIV system can be connected to two remote (DR) XIV systems in a fan-out configuration, as shown in Figure 4-19. Both remote XIV systems could be at the same location, or each of the target systems could be at a different location. Certain volumes on the local XIV system are copied to one remote XIV system, and other volumes on the same local XIV system are copied to a different remote XIV system. This configuration can be used when each XIV system at the DR site has less available capacity than the XIV system at the local site.

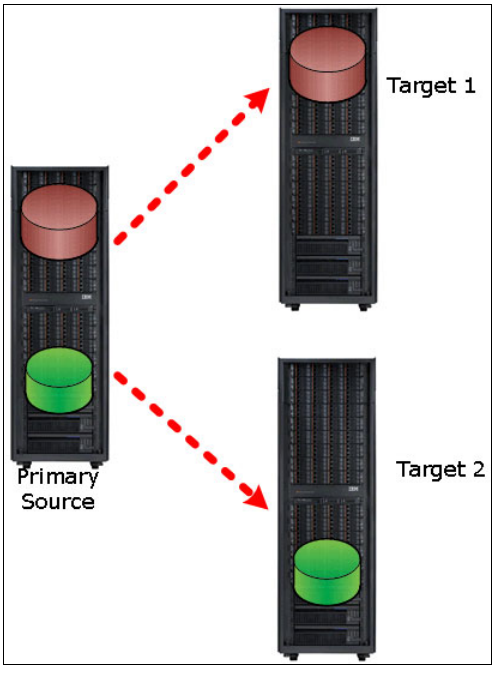


Figure 4-19 Fan-out target configuration

- XIV target configuration: synchronous and asynchronous fan-out

  XIV supports both synchronous and asynchronous mirroring (for different peers) on the same XIV system, so a single local XIV system could have certain peers synchronously mirrored to a remote XIV system at a metro distance, whereas other peers are asynchronously mirrored to a remote XIV system at a global distance, as shown in Figure 4-20. This configuration can be used when higher priority data is synchronously mirrored to another XIV system within the metro area, and lower priority data is asynchronously mirrored to an XIV system within or outside the metro area.

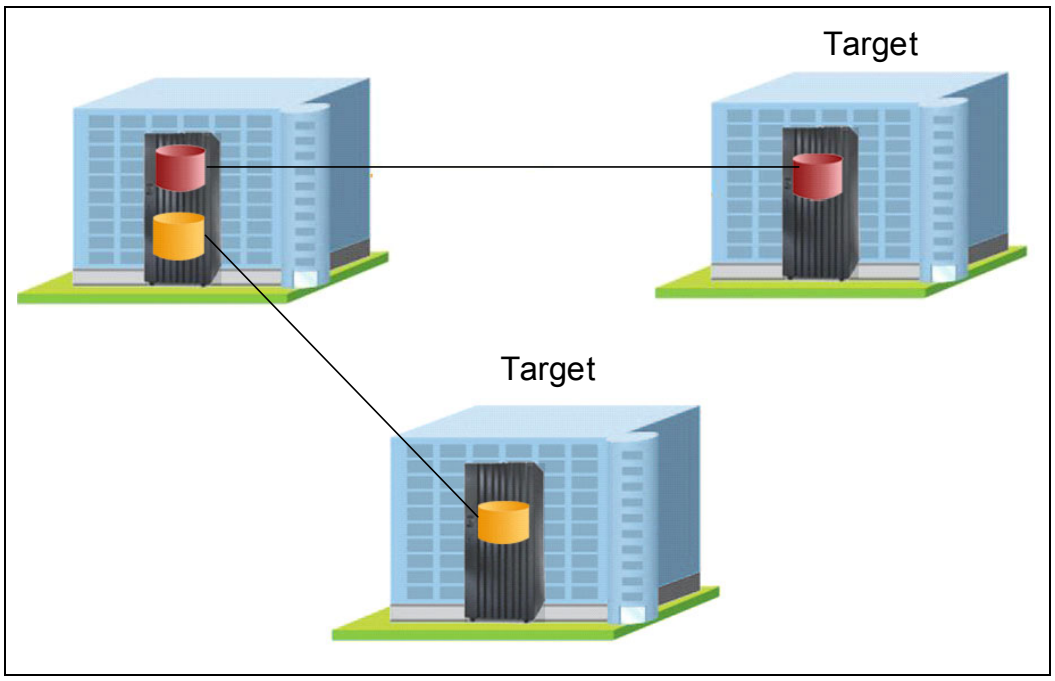


Figure 4-20 Synchronous and asynchronous fan-out

- XIV target configuration: fan-in

  Two (or more) local XIV systems can have peers mirrored to a single remote XIV system in a fan-in configuration, as shown in Figure 4-21 on page 73. This configuration must be evaluated carefully and used with caution because there is a risk of overloading the single remote XIV system. The performance capability of the single remote XIV system must be carefully reviewed before implementing a fan-in configuration.

  This configuration can be used in situations where there is a single disaster recovery data center supporting multiple production data centers, or when multiple XIV systems are mirrored to a single XIV system at a service provider.

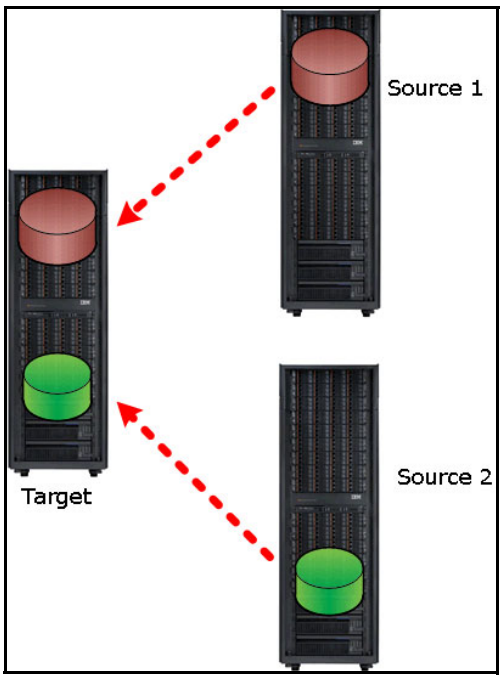


Figure 4-21 Fan-in configuration

- XIV target configuration: Bidirectional

  Two different XIV systems can have different volumes mirrored in a bidirectional configuration, as shown in Figure 4-22. This configuration can be used for situations where there are two active production sites and each site provides a DR solution for the other. Each XIV system is active as a production system for certain peers and as a mirroring target for other peers.

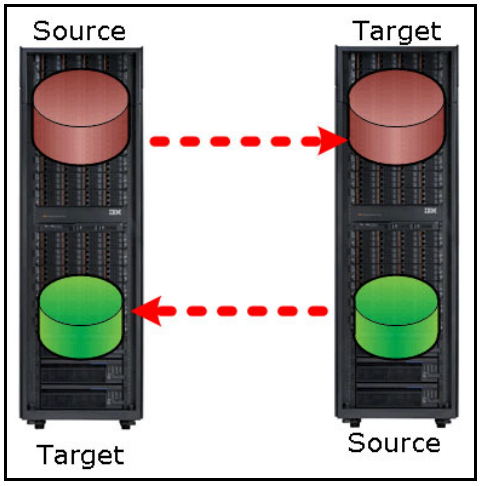


Figure 4-22 Bidirectional configuration

4.4.2 Setting the maximum initialization and synchronization rates

The XIV system allows a user-specifiable maximum rate (in MBps) for remote mirroring coupling initialization, a different user-specifiable maximum rate for normal sync jobs, and another for re-synchronization. The initialization rate, sync job rate, and resynchronization rate are specified for each mirroring target using the XCLI command **`target_config_sync_rates`**. The actual effective initialization or synchronization rate will also be dependent on the number and speed of connections between the XIV systems. The maximum initialization rate must be less than or equal to the maximum sync job rate (asynchronous mirroring only), which must be less than or equal to the maximum resynchronization rate.

Important: In normal mirror operations, the rates are cumulative; for example, if initialization, synchronous, and asynchronous operations are all active, the amount of data the XIV will attempt to send is the sum of the those three values.

The defaults are as follows:

- Maximum initialization rate: 100 MBps
- Maximum sync job: 300 MBps
- Maximum resync rate: 300 MBps

4.4.3 Connecting XIV mirroring ports

After defining remote mirroring targets, one-to-one connections must be made between ports on each XIV system. For an illustration of these actions using the GUI or the XCLI, see 4.11, “Using the GUI or XCLI for remote mirroring actions” on page 99.

- FC ports

  For the XIV Fibre Channel (FC) ports, connections are unidirectional—from an initiator port (for example, Interface Module Port 4 is configured as a Fibre Channel initiator by default) on the source XIV system to a target port (typically Interface Module Port 2) on the target XIV system. Use a minimum of four connections (two connections in each direction, from ports in two different modules, using a total of eight ports) to provide availability protection. See Figure 4-23.

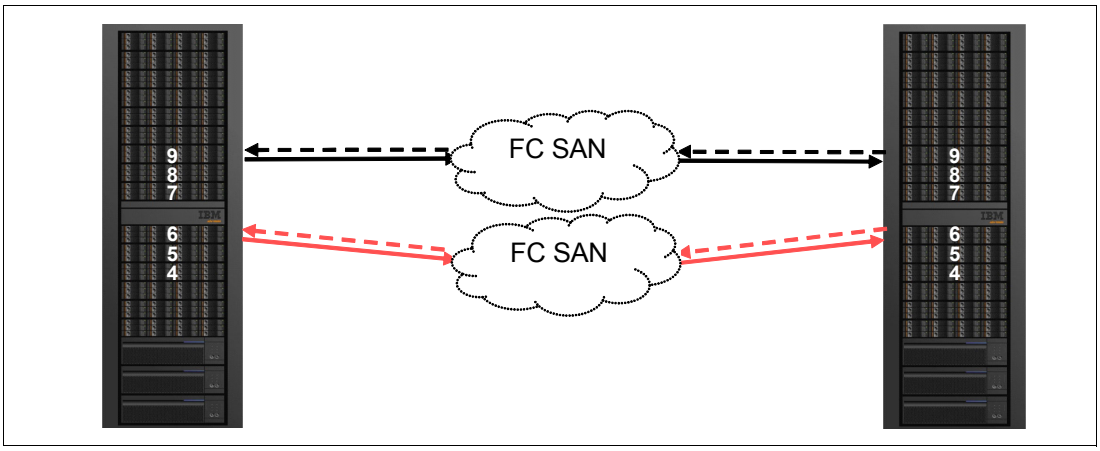


Figure 4-23 Connecting XIV mirroring ports (FC connections)

In Figure 4-23 on page 74, the solid lines represent mirroring connections used during normal operation (the mirroring target system is on the right), and the dotted lines represent mirroring connections used when production is running at the disaster recovery site and changes are being copied back to the original production site (mirroring target is on the left.)

XIV Fibre Channel ports can be easily and dynamically configured as initiator or target ports.

- iSCSI ports

For iSCSI ports, connections are bidirectional.

Important: If the IP network includes firewalls between the mirrored XIV systems, TCP port 3260 (iSCSI) must be open within firewalls so that iSCSI replication can work.

Use a minimum of two connections (with each of these ports in a different module) using a total of four ports to provide availability protection. In Figure 4-24, the solid lines represent data flow during normal operation and the dotted lines represent data flow when production is running at the disaster recovery site and changes are being copied back to the original production site.

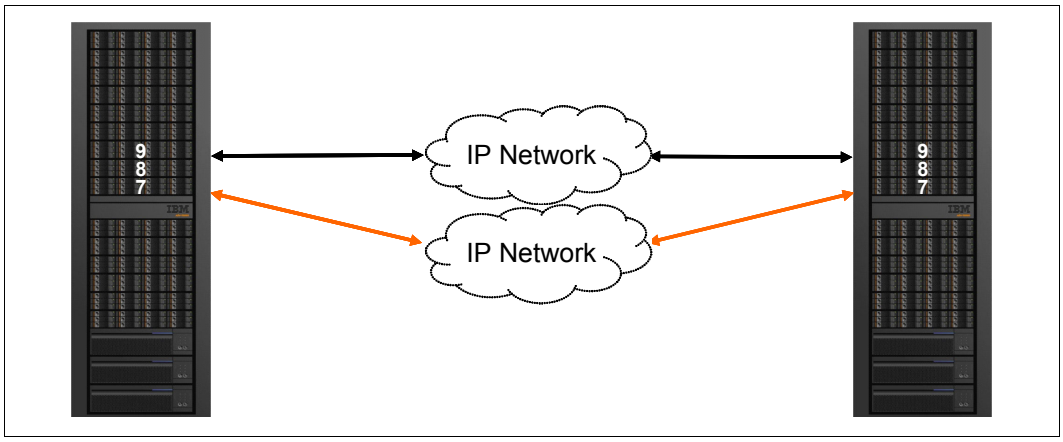


Figure 4-24 Connecting XIV mirroring ports (e connections)

Note: For asynchronous mirroring over iSCSI links, a reliable, dedicated network must be available. It requires consistent network bandwidth and a non-shared link.

4.4.4 Defining the XIV mirror coupling and peers: Volume

After the mirroring targets have been defined, a coupling or mirror can be defined, creating a mirroring relationship between two peers.

Before describing actions involved in creating mirroring pairs, the basic XIV concepts are introduced.

Storage pools, volumes, and consistency groups

An XIV storage pool is a purely administrative construct used to manage XIV logical and physical capacity allocation.

An XIV volume is a logical volume that is presented to an external server as a logical unit number (LUN). An XIV volume is allocated from logical and physical capacity within a single XIV storage pool. The physical capacity on which data for an XIV volume is stored is always spread across all available disk drives in the XIV system

The XIV system is *data aware*. It monitors and reports the amount of physical data written to a logical volume and does not copy any part of the volume that has not been used yet to store any actual data.

In Figure 4-25, seven logical volumes have been allocated from a storage pool with 40 TB of capacity. Remember that the capacity assigned to a storage pool and its volumes is spread across all available physical disk drives in the XIV system.

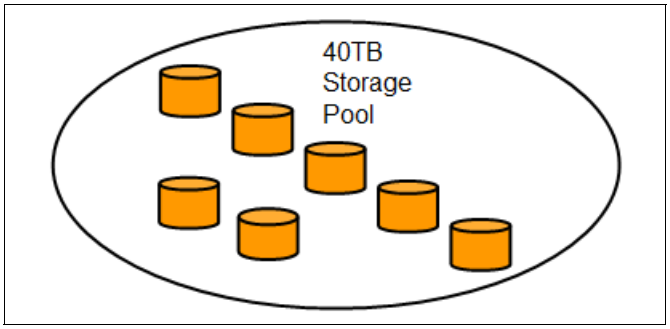


Figure 4-25 Storage pool with seven volumes

With remote mirroring, the concept of consistency group represents a logical container for a group of volumes, allowing them to be managed as a single unit. Instead of dealing with many volume remote mirror pairs individually, consistency groups simplify the handling of many pairs considerably.

An XIV consistency group exists within the boundary of an XIV storage pool in a single XIV system, which means that you can have different CGs in different storage pools within an XIV storage system, but a CG cannot span multiple storage pools. All volumes in a particular consistency group are in the same XIV storage pool.

In Figure 4-26, an XIV storage pool with 40 TB capacity contains seven logical volumes. One consistency group has been defined for the XIV storage pool, but no volumes have been added to or created in the consistency group.

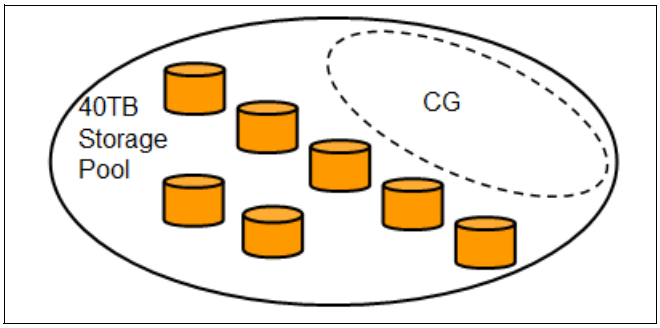


Figure 4-26 Consistency group defined

Volumes can be easily and dynamically (that is, without stopping mirroring or application I/Os) added to a consistency group.

In Figure 4-27, five of the seven existing volumes in the storage pool have been added to the consistency group in the storage pool. One or more additional volumes can be dynamically added to the consistency group at any time. Also, volumes can be dynamically moved from another storage pool to the storage pool containing the consistency group, and then added to the consistency group.

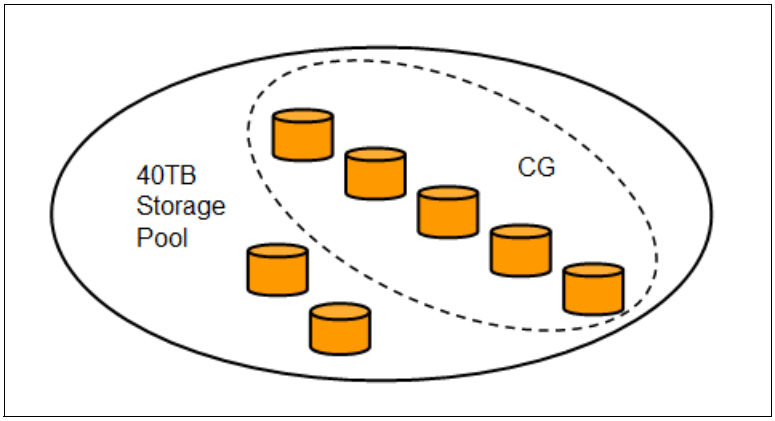


Figure 4-27 Volumes added to the consistency group

Volumes can also be easily and dynamically removed from an XIV consistency group. In Figure 4-28, one of the five volumes has been moved out of the consistency group, leaving four volumes remaining in the consistency group. It is also possible to remove all volumes from a consistency group.

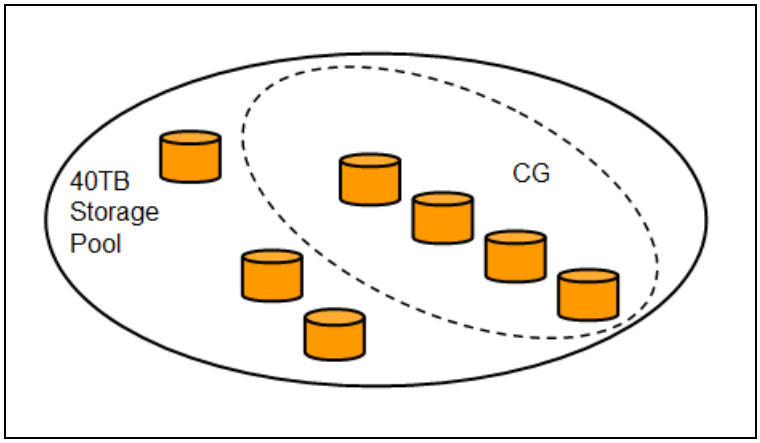


Figure 4-28 Volume removed from the consistency group

Dependent write consistency

XIV remote mirroring provides dependent write consistency, preserving the order of dependent writes in the mirrored data. Applications and databases are developed to be able to perform a fast restart from volumes that are consistent in terms of dependent writes.

Dependent writes: Normal operation

Applications or databases often manage dependent write consistency using a 3-step process such as the sequence of three writes shown in Figure 4-29. Even when the writes are directed at different logical volumes, the application ensures that the writes are committed in order during normal operation.

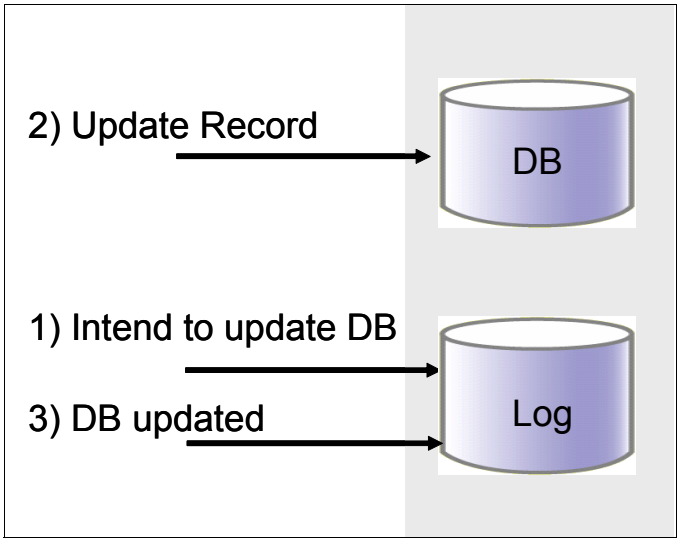


Figure 4-29 Dependent writes: Normal operation

Dependent writes: Failure scenario

In the event of a failure, applications or databases manage dependent writes, as shown in Figure 4-30. If the database record is not updated (step 2), the application does not allow *DB updated* (step 3) to be written to the log.

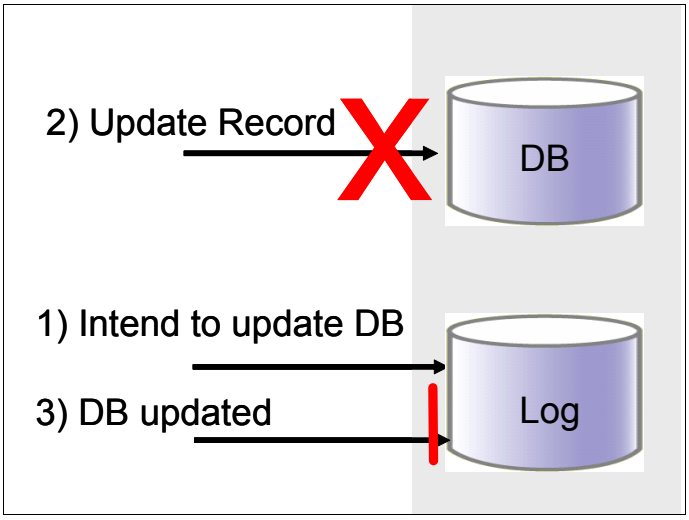


Figure 4-30 Dependent writes: Failure scenario

Just as the application or database manages dependent write consistency for the production volumes, the XIV system must manage dependent write consistency for the mirror target volumes.

If multiple volumes have dependent write activity, they can be put into a single storage pool in the XIV system and then added to an XIV consistency group to be managed as a single unit

for remote mirroring. Any mirroring actions are taken simultaneously against the mirrored consistency group as a whole, preserving dependent write consistency. Mirroring actions cannot be taken against an individual volume pair while it is part of a mirrored CG. However, an individual volume pair can be dynamically removed from the mirrored consistency group.

XIV also supports creation of application-consistent data on the remote mirroring target volumes, as described in 4.5.4, “Creating application-consistent data at both local and remote sites” on page 93.

Defining mirror coupling and peers

After the remote mirroring targets have been defined, a coupling or mirror can be defined, creating a mirroring relationship between two peers.

The two peers in the mirror coupling can be either two volumes (volume peers) or two consistency groups (CG peers), as shown in Figure 4-31.

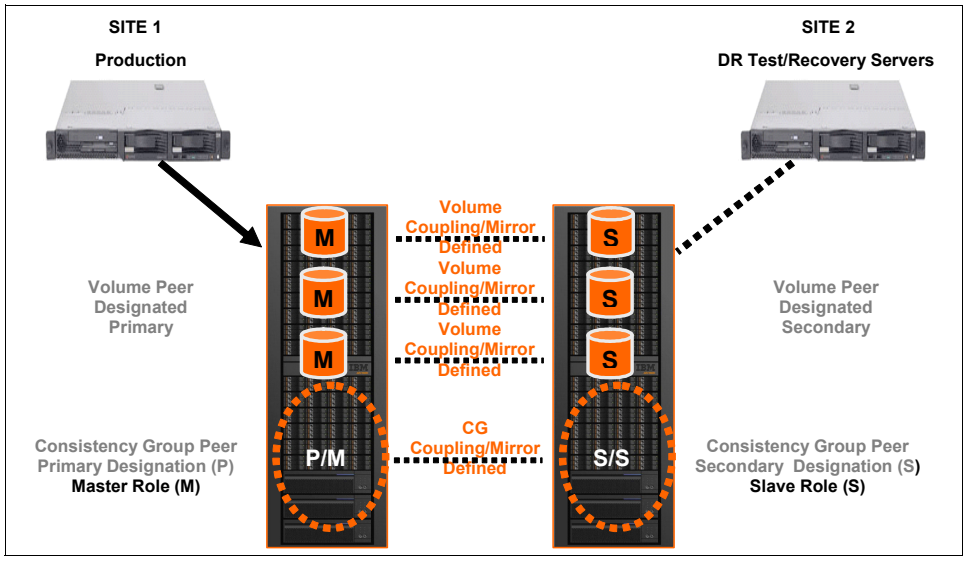


Figure 4-31 Defining mirror coupling

Each of the two peers in the mirroring relationship is given a designation and a role. The designation indicates the original or *normal* function of each of the two peers—either primary or secondary. The peer designation does not change with operational actions or commands. (If necessary, the peer designation can be changed by explicit user command or action.)

The role of a peer indicates its current (perhaps temporary) operational function (either master or slave). The operational role of a peer can change as the result of user commands or actions. Peer roles typically change during DR testing or a true disaster recovery and production site switch.

When a mirror coupling is created, the first peer specified (for example, the volumes or consistency group (CG) at site 1, as shown in Figure 4-31) is the source for data to be replicated to the target system, so it is given the primary designation and the master role.

Important: A consistency group to be mirrored must not contain any volumes when the CG coupling is defined, or the coupling will not allow it to be defined.

The second peer specified (or automatically created by the XIV system) when the mirroring coupling is created is the target of data replication, so it is given the secondary designation and the slave role.

When a mirror coupling relationship is first created, no data movement occurs.

4.4.5 Activating an XIV mirror coupling

When an XIV mirror coupling is first activated, all actual data existing on the master is either copied to the slave (normal initialization) or verified to be on the slave, and only changed data is copied (offline initialization). This process is referred to as *initialization*. XIV remote mirroring copies volume identification information (that is, physical volume ID/PVID) and any actual data on the volumes. Space that has not been used is not copied.

Initialization might take a significant amount of time if a large amount of data exists on the master when a mirror coupling is activated. As discussed earlier, the rate for this initial copy of data can be specified by the user. The speed of this initial copy of data will also be affected by the connectivity and bandwidth (number of links and link speed) between the XIV primary and secondary systems.

As an option to remove the impact of distance on initialization, XIV mirroring can be initialized with the target system installed locally, and the target system can be disconnected after initialization, shipped to the remote site and reconnected, and mirroring reactivated.

A second option to avoid the impact of distance on initialization is by using an offline initialization. The peers can either be synchronized locally and then have the DR machine moved to its remote site, or if the target system is already at a remote location with limited WAN capabilities, apply an image backup of the master volume onto the slave, and then activate the offline mirroring initialization. Beware that if a backup tape is physically transported to the remote site, it has to be an image backup. (File-level backups do not function as expected, and will result in retransmission of 100% of the data in the pairing.) The mirror pairing is defined normally, with the addition of specifying the `offline init` option when making the definition of the pairing. See also “Offline initialization” on page 156.

If a remote mirroring configuration is set up when a volume is first created (that is, before any application data has been written to the volume), initialization will be quick.

When an XIV consistency group mirror coupling is created, the CG must be empty so there is no data movement and the initialization process is extremely fast.

The mirror coupling status at the end of initialization differs for XIV synchronous mirroring and XIV asynchronous mirroring (see “Synchronous mirroring states” on page 63 and “Storage pools, volumes, and consistency groups” on page 75), but in either case, when initialization is complete, a consistent set of data exists at the remote site. See Figure 4-32 on page 81.

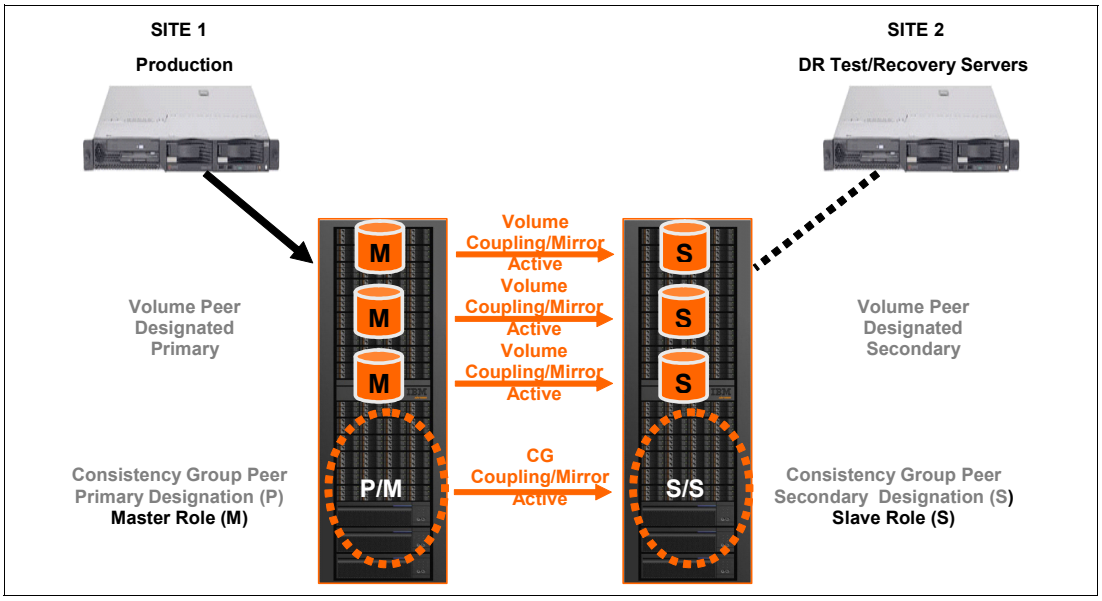


Figure 4-32 Active mirror coupling

4.4.6 Adding volume mirror coupling to consistency group mirror coupling

After a volume mirror coupling has completed initialization, the master volume can be added to a pre-existing mirrored consistency group (CG) in the same storage pool (with each mirroring type there are certain additional constraints, such as same role, target, schedule, and so on). The slave volume is automatically added to the consistency group on the remote XIV system.

In Figure 4-33, three active volume couplings that have completed initialization have been moved into the active mirrored consistency group.

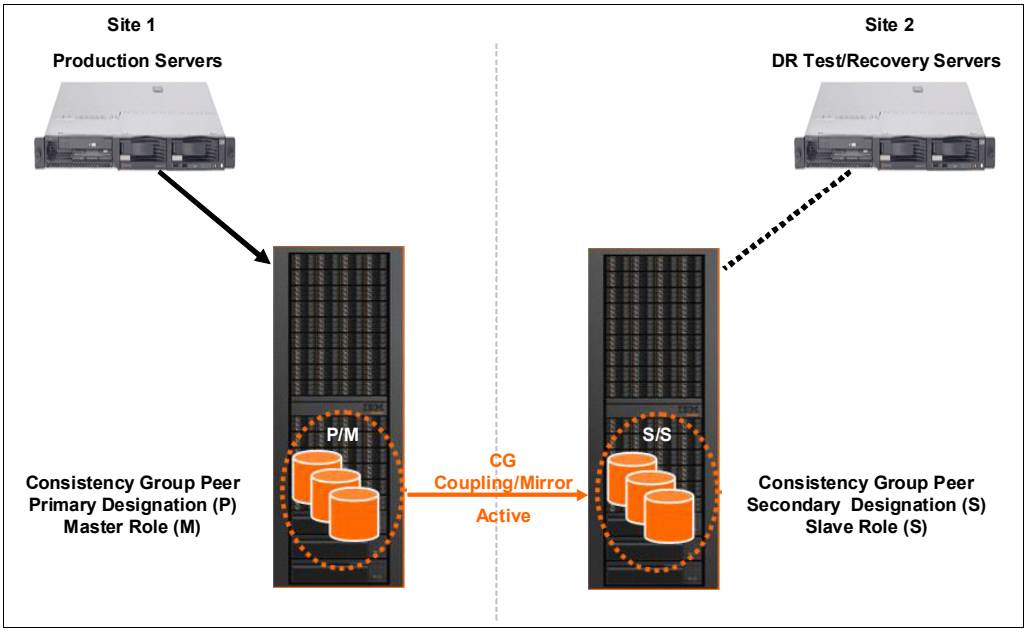


Figure 4-33 Consistency group mirror coupling

One or more additional mirrored volumes can be added to a mirrored consistency group at a later time in the same way.

It is also important to realize that in a CG all volumes have the same role. Also, consistency groups are handled as a single entity and, for example, in asynchronous mirroring, a delay in replicating a single volume affects the status of the entire CG.

4.4.7 Normal operation: Volume mirror coupling and CG mirror coupling

XIV mirroring *normal operation* begins after initialization has completed successfully and all actual data on the master volume at the time of activation has been copied to the slave volume. During normal operation, a consistent set of data is available on the slave volumes.

Normal operation, statuses, and reporting differ for XIV synchronous mirroring and XIV asynchronous mirroring. See Chapter 5, “Synchronous Remote Mirroring” on page 113, and Chapter 6, “Asynchronous remote mirroring” on page 149, for details.

During normal operation, a single XIV system can contain one or more mirrors of volume peers and also one or more mirrors of CG peers, as shown in Figure 4-34.

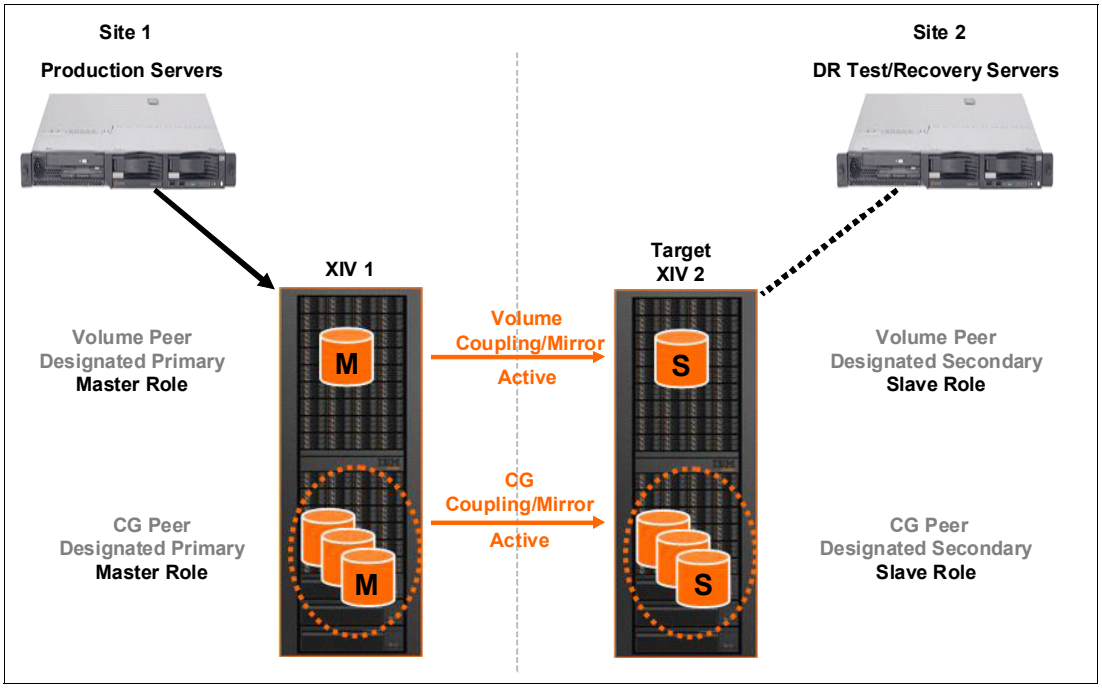


Figure 4-34 Normal operations: Volume mirror coupling and CG mirror coupling

4.4.8 Deactivating XIV mirror coupling: Change recording

An XIV mirror coupling can be deactivated by a user command. In this case, the mirror transitions to standby mode, as shown in Figure 4-35.

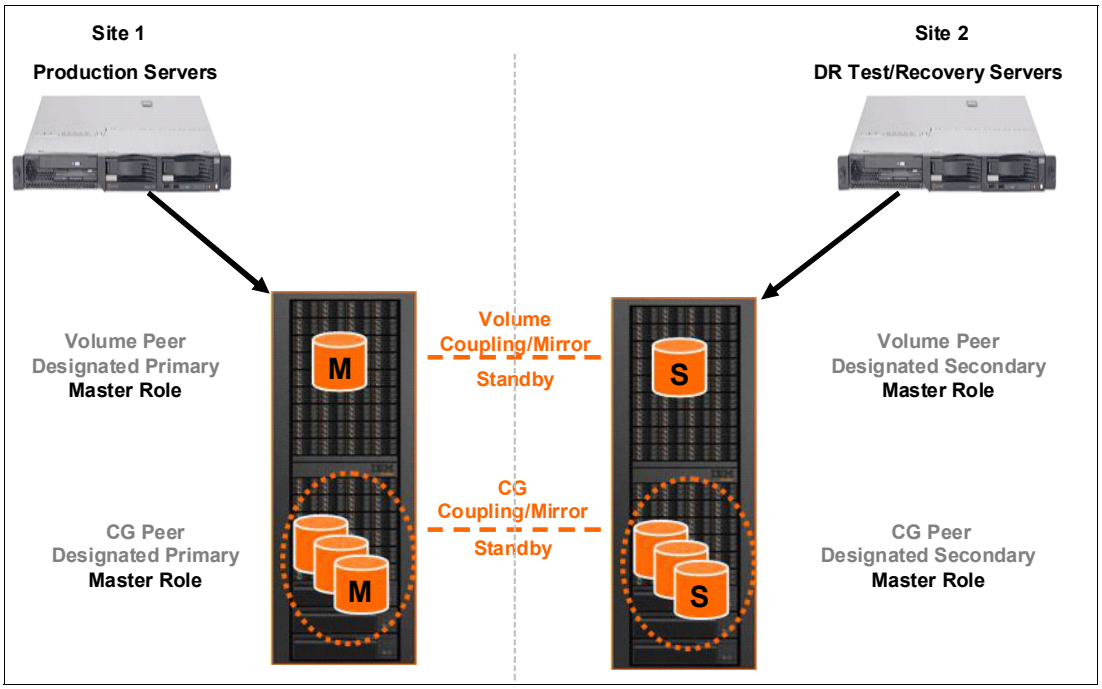


Figure 4-35 Deactivating XIV mirror coupling: Change recording

During standby mode, a consistent set of data is available at the remote site (site 2, in our example). The currency of the consistent data ages in comparison to the master volumes, and the gap increases while mirroring is in standby mode.

In synchronous mirroring, during standby mode, XIV metadata is used to note which parts of a master volume have changed but have not yet been replicated to the slave volume (because mirroring is not currently active). The actual changed data is not retained in cache, so there is no danger of exhausting cache while mirroring is in standby mode.

When synchronous mirroring is reactivated by a user command or communication is restored, the metadata is used to resynchronize changes from the master volumes to the slave volumes. XIV mirroring records changes for master volumes only. If it is desirable to record changes to both peer volumes while mirroring is in standby mode, the slave volume must be changed to a master volume.

In asynchronous mirroring, metadata is not used and the comparison between the most-recent and last-replicated snapshots indicates the data that must be replicated.

Planned deactivation of XIV remote mirroring can be done to suspend remote mirroring during a planned network outage or DR test, or to reduce bandwidth during peak load.

4.4.9 Changing role of slave volume or CG

When XIV mirroring is active, the slave volume or CG is locked and write access is prohibited. To allow write access to a slave peer, in case of failure or unavailability of the master, the slave volume role must be changed to the master role. See Figure 4-36.

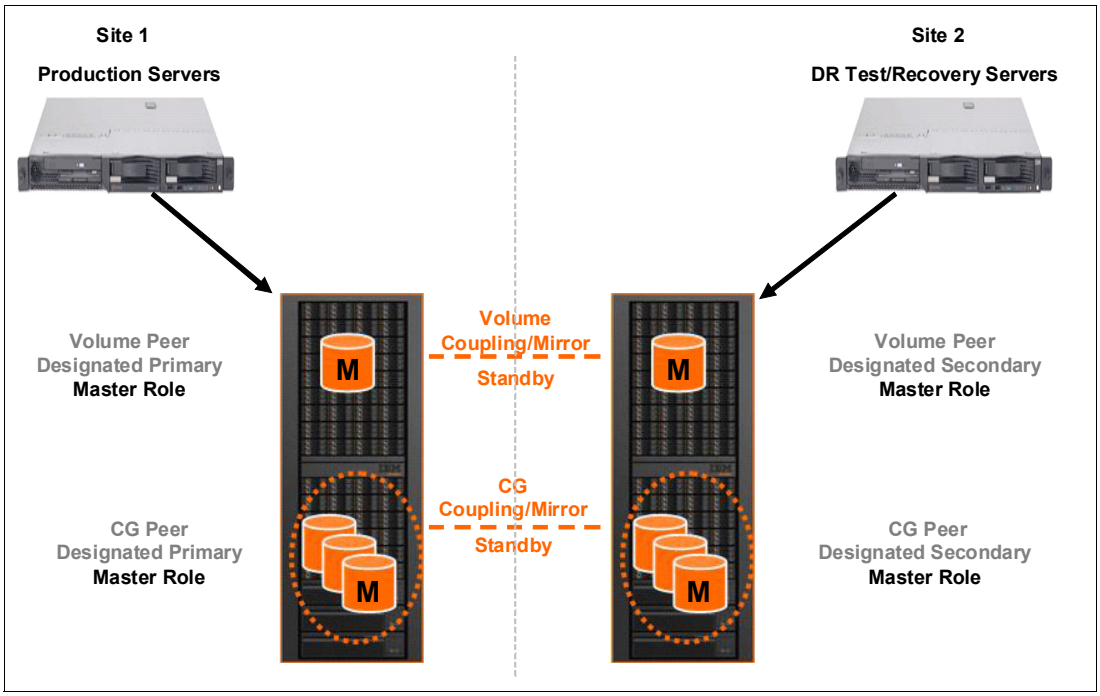


Figure 4-36 Changing role of slave volume or CG

Changing the role of a volume from slave to master allows the volume to be accessed. In synchronous mirroring, changing the role also starts metadata recording for any changes made to the volume. This metadata can be used for resynchronization (if the new master volume remains the master when remote mirroring is reactivated). In asynchronous mirroring, changing a peer's role automatically reverts the peer to its last-replicated snapshot.

When mirroring is in standby mode, both volumes might have the master role, as shown in the following section. When changing roles, both peer roles must be changed if possible (the exception being a site disaster or complete system failure). Changing the role of a slave volume or CG is typical during a true disaster recovery and production site switch.

4.4.10 Changing role of master volume or CG

During a true disaster recovery, to resume production at the remote site a slave must have its role changed to the master role.

In synchronous mirroring, changing a peer role from master to slave allows the slave to accept mirrored data from the master and cause deletion of metadata that was used to record any changes while the peer had the master role.

In asynchronous mirroring, changing a peer's role automatically reverts the peer to its last-replicated snapshot. If at any point in time the command is run on the slave (changing the slave to a master), the former master must first be changed to the slave role (upon recovery of the primary site) *before* changing the secondary role back from master to slave.

Both peers might temporarily have the master role when a failure at site 1 has resulted in a true disaster recovery production site switch from site 1 to site 2. When site 1 becomes available again and there is a requirement to switch production back to site 1, the production changes made to the volumes at site 2 must be resynchronized to the volumes at site 1. To do this, the peers at site 1 must change their role from master to slave, as shown in Figure 4-37.

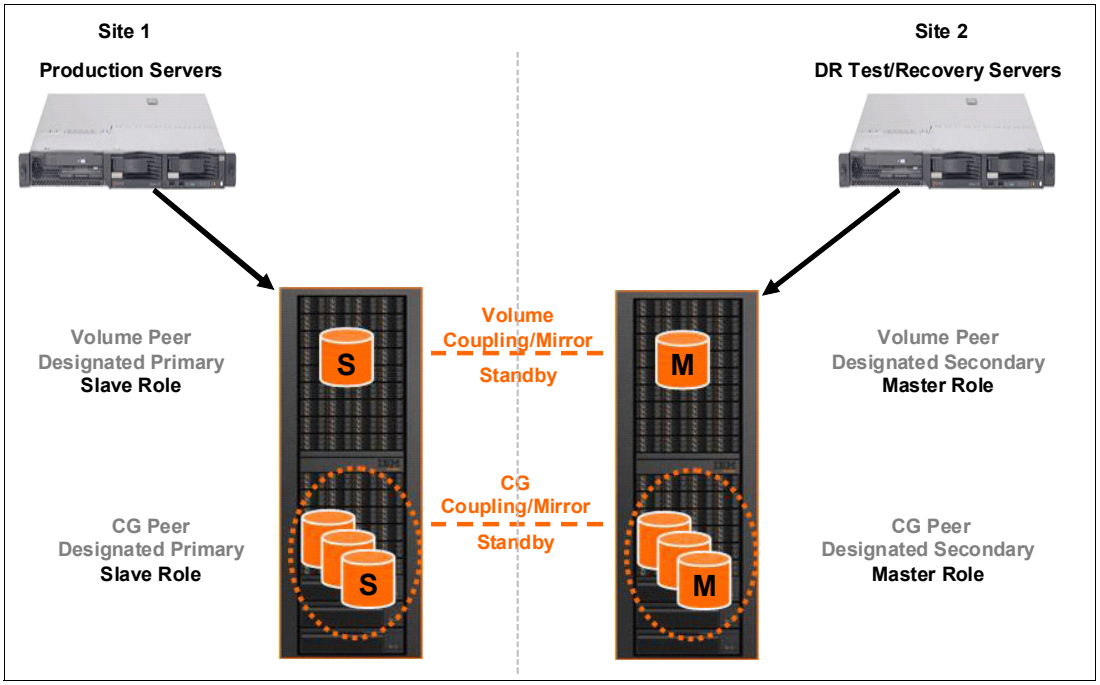


Figure 4-37 Changing role to slave volume and CG

4.4.11 Mirror reactivation and resynchronization: Normal direction

In synchronous mirroring, when mirroring has been in standby mode, any changes to volumes with the master role were recorded in metadata. Then when mirroring is reactivated, changes recorded in metadata for the current master volumes are resynchronized to the current slave volumes. See Figure 4-38.

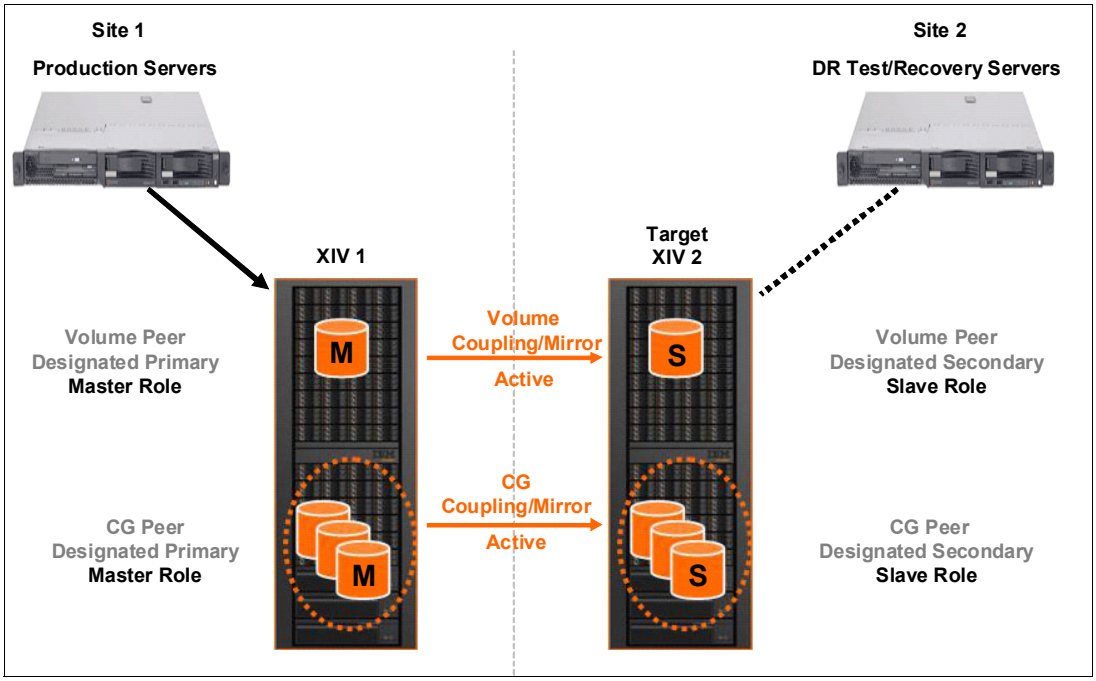


Figure 4-38 Mirror reactivation and resynchronization: Normal direction

The rate for this resynchronization of changes can be specified by the user in MBps using the XCLI **`target_config_sync_rates`** command.

When XIV mirroring is reactivated in the normal direction, changes recorded at the primary peers are copied to the secondary peers.

The following examples are of mirror deactivation and reactivation in the same direction:

- Remote mirroring is temporarily inactivated because of communication failure and then automatically reactivated by the XIV system when communication is restored.
- Remote mirroring is temporarily inactivated to create an extra copy of consistent data at the secondary.
- Remote mirroring is temporarily inactivated by user action during peak load in an environment with constrained network bandwidth.

4.4.12 Synchronous mirror deletion and using offline initialization for resynchronization

Starting with version 11.4, synchronous mirroring can use offline initialization (which was available before, only for asynchronous mirroring). When a mirror is suspended for a long time, one can consider the deletion of the mirror and avoid the tracking of the changed data for that long period. After the mirror is re-established, it should use the offline initialization option to minimize the data transfer to the changed data alone.

4.4.13 Reactivation, resynchronization, and reverse direction

When XIV mirroring is reactivated in the reverse direction, as shown in the previous section, changes recorded at the secondary peers are copied to the primary peers. The primary peers must change the role from master to slave before mirroring can be reactivated in the reverse direction. See Figure 4-39.

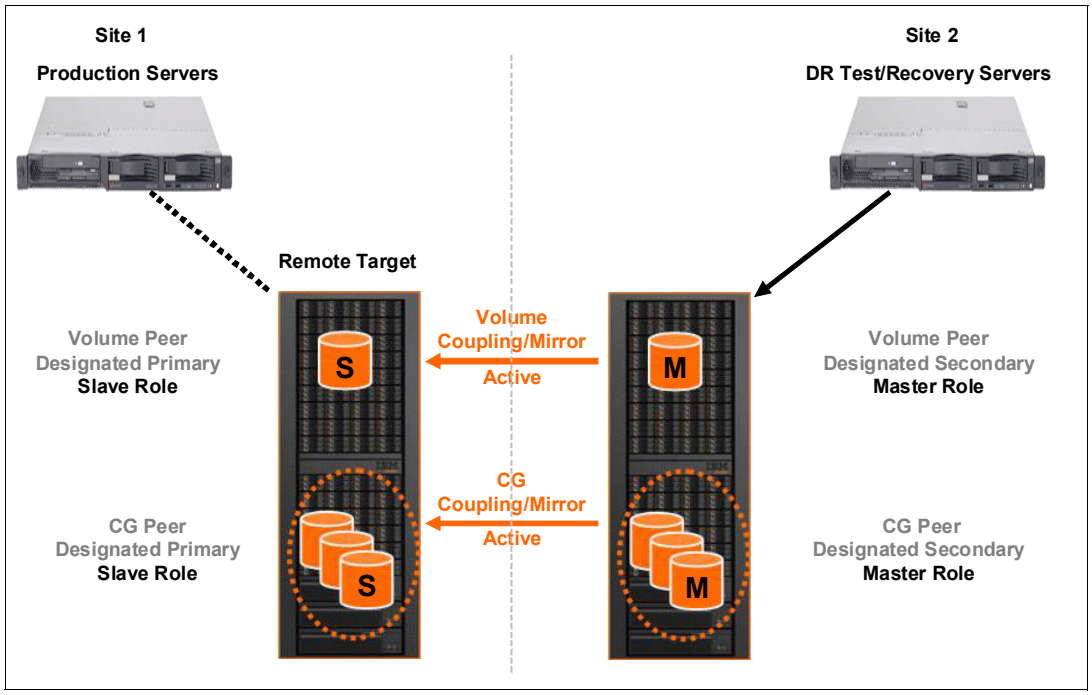


Figure 4-39 Reactivation and resynchronization

A typical usage example of this scenario is when returning to the primary site after a true disaster recovery with production switched to the secondary peers at the remote site.

4.4.14 Switching roles of mirrored volumes or CGs

When mirroring is active and synchronized (consistent), the master and slave roles of mirrored volumes or consistency groups can be switched simultaneously. Role switching is typical for returning mirroring to the normal direction after changes have been mirrored in the reverse direction after a production site switch. Role switching is also typical for any planned production site switch. Host server write activity and replication activity must be paused briefly before and during the role switch. Additionally, in the case of asynchronous mirroring, at least one sync job must complete, prior to the switch, to ensure the expected point in time copy of the data exists.

4.4.15 Adding a mirrored volume to a mirrored consistency group

First, make sure that the following constraints are respected:

- Volume and CG must be associated with the same pool
- Volume is not already part of a CG
- Command must be issued only on the master CG
- Command must not be run during initialization of volume or CG

- The volume mirroring settings must be identical to those of the CG:
  - Mirroring type
  - Mirroring role
  - Mirroring status
  - Mirroring target
  - Target pool
- Both volume synchronization status and mirrored CG synchronization status are either RPO OK for asynchronous mirroring or Synchronized for synchronous mirroring.

To add a volume mirror to a mirrored consistency group (for instance, when an application needs additional capacity), use the following steps:

1. Define XIV volume mirror coupling from the additional master volume at XIV 1 to the slave volume at XIV 2.
2. Activate XIV remote mirroring from the additional master volume at XIV 1 to the slave volume at XIV 2.
3. Monitor initialization until it is complete. Volume coupling initialization must complete before the coupling can be moved to a mirrored CG.
4. Add the additional master volume at XIV 1 to the master consistency group at XIV 1. (The additional slave volume at XIV 2 will be automatically added to the slave consistency group at XIV 2.)

In Figure 4-40, one volume has been added to the mirrored XIV consistency group. The volumes must be in a volume peer relationship and must have completed initialization.

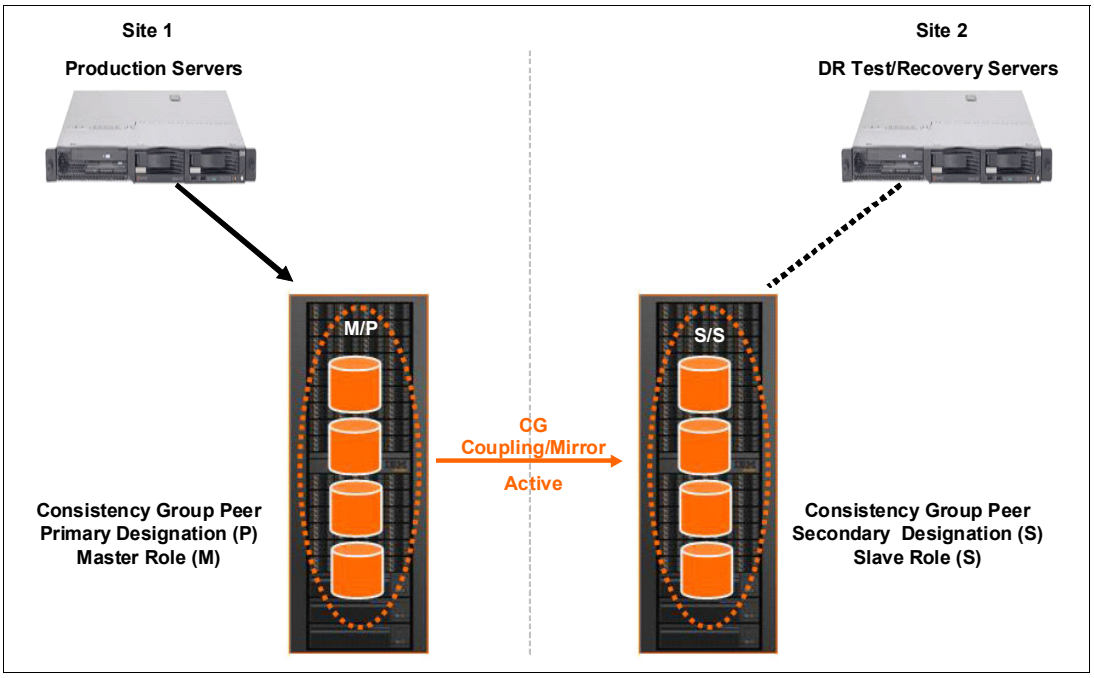


Figure 4-40 Adding a mirrored volume to a mirrored consistency group

For further details, see the following sections:

- 4.4.4, “Defining the XIV mirror coupling and peers: Volume” on page 75
- 4.4.6, “Adding volume mirror coupling to consistency group mirror coupling” on page 81

4.4.16 Removing a mirrored volume from a mirrored consistency group

If a volume in a mirrored consistency group is no longer being used by an application or if actions must be taken against the individual volume, it can be dynamically removed from the consistency group.

To remove a volume mirror from a mirrored consistency group, use the following steps:

1. Remove the master volume from the master consistency group at site 1. (The slave volume at site 2 will be automatically removed from the slave CG.)
2. When a mirrored volume is removed from a mirrored CG, it retains its mirroring status and settings and continues remote mirroring until deactivated.

In Figure 4-41, one volume has been removed from the example mirrored XIV consistency group with three volumes. After being removed from the mirrored CG, a volume will continue to be mirrored as part of a volume peer relationship.

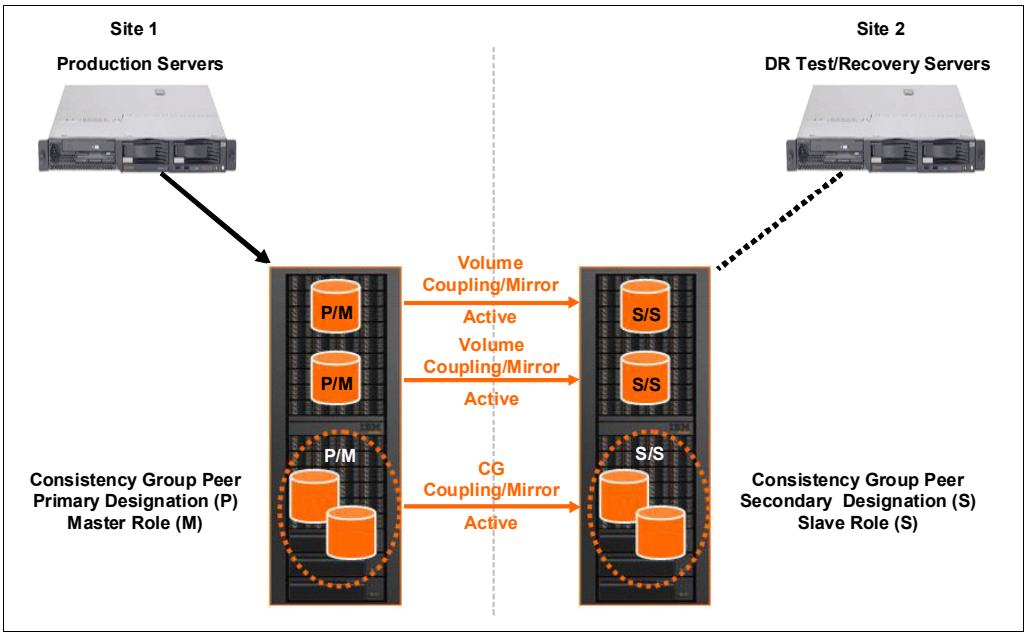


Figure 4-41 Removing a mirrored volume from a mirrored CG

4.4.17 Deleting mirror coupling definitions

When an XIV mirror coupling is deleted, all metadata and mirroring definitions are deleted, and the peers do not have any relationship at all (Figure 4-42). However, any volumes and consistency groups mirroring snapshots remain on the local and remote XIV systems. To restart XIV mirroring, it is possible to use offline initialization instead of a full copy of data.

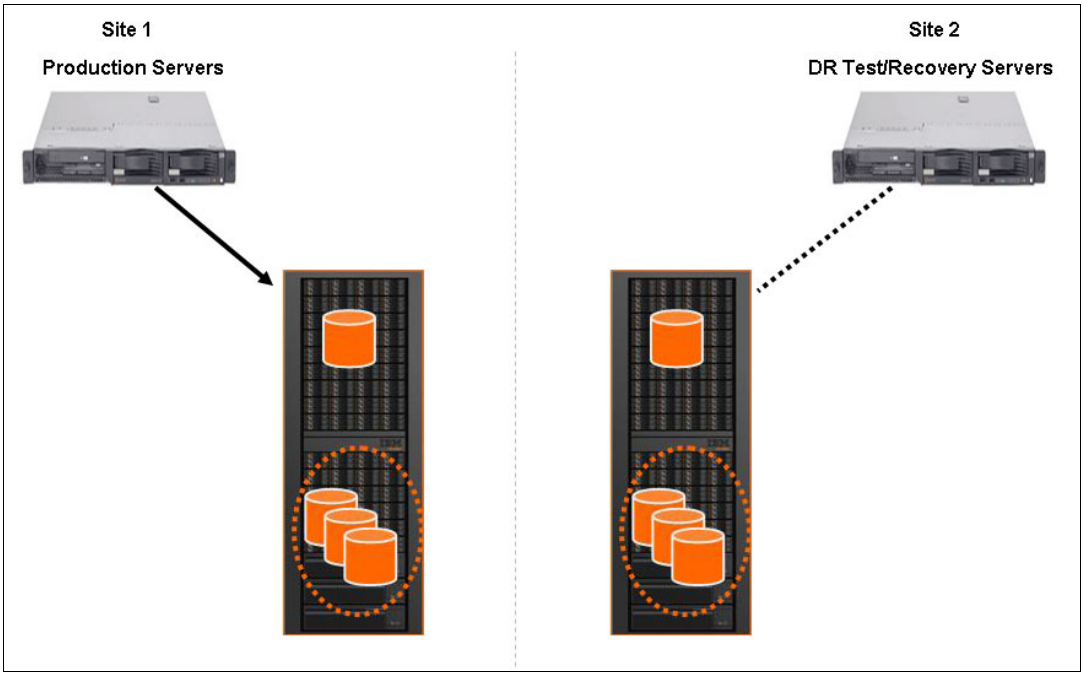


Figure 4-42 Deleting mirror coupling definitions

Typical usage of mirror deletion is a one-time data migration using remote mirroring. This includes deleting the XIV mirror couplings after the migration is complete.

4.5 Best practice usage scenarios

The following best practice usage scenarios begin with the *normal operation* remote mirroring environment shown in Figure 4-43.

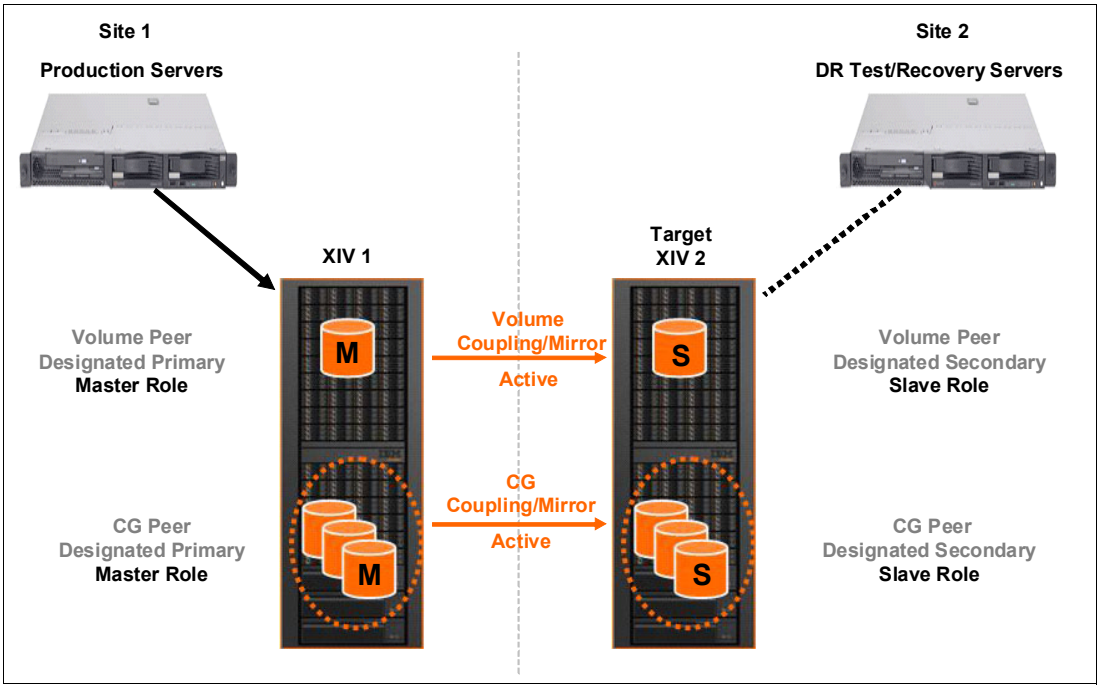


Figure 4-43 Remote mirroring environment for scenarios

4.5.1 Failure at primary site: Switch production to secondary

This scenario begins with normal operation of XIV remote mirroring from XIV 1 to XIV 2.The failure happens in XIV 1.The assumption is that the failure nature is one that does not affect the data. Thus, the data on the XIV 1 might be stale, yet it is consistent and readily available for resynchronization between the two peers, when the XIV is repaired.

1. XIV remote mirroring might have been deactivated by the failure.
2. Change the role of the peer at XIV 2 from slave to master. This allows the peer to be accessed for writes from a host server, and also triggers recording of any changes of metadata for synchronous mirroring. For asynchronous mirroring, changing the role from slave to master causes the last-replicated snapshot to be restored to the volume. Now both XIV 1 and XIV 2 peers have the master role.
3. Map the master (secondary) peers at XIV 2 to the DR servers.
4. Bring the DR servers online to begin production using the XIV 2.
5. When the failure at XIV 1 has been corrected and XIV 1 is available, deactivate mirrors at XIV 1 if they are not already inactive.
6. Unmap XIV 1 peers from servers if necessary.
7. Change the role of the peer at XIV 1 from master to slave.
8. Activate remote mirroring from the master peers at XIV 2 to the slave peers at XIV 1. This starts resynchronization of production changes from XIV 2 to XIV 1. Alternately, or if the mirror pair was deleted, offline initialization can be used instead of the resynchronization to accomplish the same result.

9. Monitor the progress to ensure that resynchronization is complete.
10. Quiesce production applications at XIV 2 to ensure that application-consistent data is copied to XIV 1.
11. Unmap master peers at XIV 2 from DR servers.
12. For asynchronous mirroring, monitor completion of sync job and change the replication interval to never.
13. Monitor to ensure that no more data is flowing from XIV 2 to XIV 1.
14. Switch roles of master and slave. XIV 1 peers now have the master role and XIV 2 peers now have the slave role.
15. For asynchronous mirroring, change the replication schedule to the desired interval.
16. Map master peers at XIV 1 to the production servers.
17. Bring production servers online using XIV 1.

4.5.2 Complete destruction of XIV 1

This scenario begins with normal operation of XIV remote mirroring from XIV 1 to XIV 2, followed by complete destruction of XIV 1:

1. Change the role of the peer at XIV 2 from slave to master. This allows the peer to be accessed for writes from a host server.
2. Map the new master peer at XIV 2 to the DR servers.
3. Bring the DR servers online to begin to begin production workload using XIV 2.
4. Deactivate XIV remote mirroring from the master peer at XIV 2 if necessary (It might have already been deactivated as a result of the XIV 1 failure).
5. Delete XIV remote mirroring from the master peer at XIV 2.
6. Rebuild XIV 1, including configuration of the new XIV system at XIV 1, the definition of remote targets for both XIV 1 and XIV 2, and the definition of connectivity between XIV 1 and XIV 2.
7. Define XIV remote mirroring from the master peer at XIV 2 to the slave peer at XIV 1.
8. Activate XIV remote mirroring from the master peer at XIV 2 to the slave peer at XIV 1. This results in a full copy of all actual data on the master peer at XIV 2 to the slave volume at XIV 1. Depends on the extent of the XIV 1 destruction, the case might be that the original LUNs are still available on it and, as such an offline initialization can be used to expedite the process.
9. Monitor initialization until it is complete.
10. Quiesce the production applications at XIV 2 to ensure that all application-consistent data is copied to XIV 1.
11. Unmap master peers at XIV 2 from DR servers.
12. For asynchronous mirroring, monitor completion of the sync job and change the replication interval to never.
13. Monitor to ensure that no more data is flowing from XIV 2 to XIV 1.
14. You can do a switch roles, which simultaneously changes the role of the peers at XIV 1 from slave to master and changes the role of the peers at XIV 2 from master to slave.

15. For asynchronous mirroring, change the replication schedule to the desired interval.
16. Map master peers at XIV 1 to the production servers.
17. Bring the servers at the primary site online and use the XIV 1 for production.

4.5.3 Using an extra copy for DR tests

This scenario begins with normal operation of XIV remote mirroring from XIV 1 to XIV 2:

1. Create a Snapshot or volume copy of the consistent data at XIV 2. The procedure differs slightly for XIV synchronous mirroring and XIV asynchronous mirroring. For asynchronous mirroring, consistent data is on the last-replicated snapshot.
2. Unlock the snapshot or volume copy.
3. Map the snapshot/volume copy to DR servers at XIV 2.
4. Bring the server at the DR site online and use the snapshot/volume copy at XIV 2 for disaster recovery testing.
5. When DR testing is complete, unmap the snapshot/volume copy from XIV 2 DR servers.
6. Delete the snapshot/volume copy if you want.

4.5.4 Creating application-consistent data at both local and remote sites

This scenario begins with normal operation of XIV remote mirroring from XIV 1 to XIV 2. This scenario can be used when the fastest possible application restart is required. Use the following procedure:

1. Mirroring is not affected; no actions are taken to change XIV remote mirroring.
2. Briefly quiesce the application using XIV 1. If the application is a database, put the database into online backup mode.
3. Ensure that all data has been copied from the master peer at XIV 1 to the slave peer at XIV 2.
4. Issue **`mirror_create_snapshot`** command at the master peer. This command creates two identical application-consistent snapshots. This is the only way to create an application-consistent snapshot on the secondary machine. The side effect is that the primary also has a copy of the snapshot. The snapshot on the primary might be used if the testing on the secondary is successful.
5. Resume normal operation of the application or database at XIV 1.
6. Unlock the snapshot.
7. Map the snapshot to DR servers at XIV 2.
8. Bring the servers at the secondary site online to begin testing, using the snapshot on XIV 2.
9. When DR testing or other use is complete, unmap the snapshot copy from XIV 2 DR servers.
10. Delete the snapshot/volume copy if you want.

4.5.5 Migration through mirroring

A migration scenario involves a one-time movement of data from one XIV system to another (for example, migration to new XIV hardware.) This scenario begins with existing connectivity between XIV 1 and XIV 2. Use the following procedure:

1. Define XIV remote mirroring from the master volume at XIV 1 to the slave volume at XIV 2.
2. Activate XIV remote mirroring.
3. Monitor initialization until it is complete.
4. Deactivate XIV remote mirroring from the master volume at XIV 1, to the slave volume at XIV 2.
5. Delete XIV remote mirroring from the master volume at XIV 1 to the slave volume at XIV 2.
6. Remove the connectivity between the XIV 1 and XIV 2 systems.
7. Redeploy the XIV system at XIV 1, if you want.

4.5.6 Migration using Hyper-Scale Mobility

The Hyper-Scale Mobility feature is described in *IBM Hyper-Scale in XIV Storage,* REDP-5053. The Hyper-Scale Mobility can be an alternative to the mirroring-based migration (described in 4.5.5, “Migration through mirroring” on page 94).

4.5.7 Adding data corruption protection to disaster recovery protection

This scenario begins with normal operation of XIV remote mirroring from XIV 1 to XIV 2 followed by creation of an additional snapshot of the master volume at XIV 1 to be used in the event of application data corruption. To create a dependent-write consistent snapshot, no changes are required to XIV remote mirroring. Use the following procedure:

1. Periodically issue **`mirror_create_snapshot`** at the master peer. This creates additional identical snapshots, one on the master and another on the slave.
2. When production data corruption is discovered, quiesce the application and take any steps necessary to prepare the application to be restored.
3. Deactivate and delete mirroring.
4. Restore production volumes from the appropriate snapshots.
5. Bring production volumes online and begin production access.
6. Remove remote volumes from the consistency group.
7. Delete or format remote volumes.
8. Delete any mirroring snapshots that exist at the production site.
9. Remove production volumes from the consistency group.
10. Define and activate mirroring optionally using the offline flag to expedite the process. Initialization results in a full copy of data.

If you want an application-consistent snapshot, use the following alternative procedure:

1. Periodically quiesce the application (or place into hot backup mode).
2. Create a snapshot of the production data at XIV 1. The procedure might be slightly different for XIV synchronous mirroring and XIV asynchronous mirroring. For asynchronous mirroring, a duplicate snapshot or a volume copy of the last-replicated snapshot can be used.

3. As soon as the snapshot or volume copy relationship has been created, resume normal operation of the application.
4. When production data corruption is discovered, deactivate mirroring.
5. Remove master peers from the consistency group on XIV 1 if necessary. (Slave peers will be automatically removed from the consistency group at XIV 2.)
6. Delete mirroring.
7. Restore the production volume from the snapshot or volume copy at XIV 1.
8. Delete any remaining mirroring-related snapshots or snapshot groups at XIV 1.
9. Delete secondary volumes at XIV 2.
10. Remove XIV 1 volumes (primary) from the consistency group.
11. Define remote mirroring peers from XIV 1 to XIV 2 (optionally using the offline initialization flag).
12. Activate remote mirroring peers from XIV 1 to XIV 2 (full copy is required).

4.5.8 Communication failure between mirrored XIV systems

This scenario begins with normal operation of XIV remote mirroring from XIV 1 to XIV 2, followed by a failure in the communication network used for XIV remote mirroring from XIV 1 to XIV 2. Use the following procedure:

1. No action is required to change XIV remote mirroring.
2. When communication between the two XIV systems is unavailable, XIV remote mirroring is automatically deactivated and changes to the master volume are recorded in metadata.
3. When communication between the XIV systems at XIV 1 and XIV 2 is restored, XIV mirroring is automatically reactivated, resynchronizing changes from the master XIV 1 to the slave XIV 2.

In the case of an extended outage, and a heavily provisioned master XIV system, the master XIV system might not have enough free space to sustain the changes' rate of host writes coming to it, and might automatically delete its most-recent and last-replicated snapshots.

If this occurs, change-tracking is effectively lost between the master and slave. To recover from this scenario, deletion of the mirror pairings and reinitialization with offline_init will provide the most timely recovery.

4.5.9 Temporary deactivation and reactivation

This scenario begins with normal operation of XIV remote mirroring from XIV 1 to XIV 2, followed by user deactivation of XIV remote mirroring for a period of time. This scenario can be used to temporarily suspend XIV remote mirroring during a period of peak activity if there is not enough bandwidth to handle the peak load or if the response time impact during peak activity is unacceptable.

1. Deactivate XIV remote mirroring from the master volume at XIV 1 to the slave volume on XIV 2. Changes to the master volume at XIV 1 will be recorded in metadata for synchronous mirroring.
2. Wait until it is acceptable to reactivate mirroring.
3. Reactivate XIV remote mirroring from the master volume at XIV 1 to the slave volume at XIV 2.

4.5.10 Connectivity type change

In cases where replication connectivity must be changed from Fibre Channel to iSCSI or from iSCSI to Fibre Channel, the XIV offline initialization feature can be used. The steps are as follows:

1. Deactivate all the mirror pairings between the master and slave systems. (This process cannot be completed if any mirrors still exist between the two systems.) Document the RPO and interval information (in case of asynchronous mirroring) if necessary for later re-creation.
2. Delete all mirror pairings between the master and slave XIV systems.
3. Delete all connectivity between the master and slave XIV systems by selecting and deleting any Fibre Channel or iSCSI paths.
4. Delete the target system definition from the mirroring environment.
5. Make any network changes required to support the new connectivity, and verify connectivity appropriately.
6. Select **Add Target** and re-create the slave system definition utilizing the desired connectivity (Fibre Channel or iSCSI).
7. Redefine connectivity appropriately (either Fibre Channel or iSCSI), maintaining paths between two modules on the master and slave XIV systems (at a minimum to maintain redundancy) and verify that the paths turn green, indicating connectivity.
8. Use XCLI and enter the t**arget_change_sync_rates** command to set any appropriate link throttles deemed necessary.
9. Redefine the mirror pairings between the master and slave volumes and select **Offline Init** on the initialization panel (see Figure 5-3 on page 116). In the cae of asynchronous mirroring, also apply RPO and interval information as documented, or making appropriate changes if needed.
10. Activate the volumes and wait for the compare and delta data transfer to complete and the volumes to have the RPO_OK status (in the case of asynchronous mirroring) and Synchronized status (in the case of synchronous mirroring).

4.5.11 Mirror type conversion

If migrating between synchronous and asynchronous replication (and asynchronous and synchronous) is necessary, offline initialization can be used as an alternative to change mirror types with minimal impact. Use the following steps:

1. Deactivate the incumbent relevant mirror pairings.
2. If the current mirroring type is Asynchronous then, document the RPO and interval information, if necessary for later re-creation.
3. Delete the relevant mirror pairs.
4. Unlock any volumes on the slave that might still be in a locked condition; if any mirror-related snapshots exist, delete them also.
5. Redefine the mirror pairings between the master and target volumes and select **Offline Init** on the initialization panel (see Figure 5-3 on page 116).
6. Enter the RPO you want, and schedule information if new mirror type is asynchronous.
7. Activate the mirror pairings, and wait for the compare and delta data to transfer, and the volumes to have the RPO_OK status (for asynchronous mirror) and the Synchronized status (for synchronous mirroring).

4.5.12 Volume resizing across asynchronous XIV mirror pairs

Because of the loose coupling between master and slave volumes in an asynchronous mirroring relationship, XIV does not support volume resizing across a defined and active mirror pair.

To resize a mirrored asynchronous volume without full reinitialization, offline init can be used. The steps are as follows:

1. Deactivate the volume pairings to be resized.
2. Delete the volume pairings to be resized.
3. Unlock the remaining volumes on the target side.
4. Remove any mirror-related snapshots (most-recent and last-replicated) from both the master and slave XIV systems.
5. Resize the volume on the master XIV system. Identically resize the volume on the slave XIV system.
6. Redefine the mirror pairing between the master and target volumes and select **Offline Init** on the initialization panel (see Figure 5-3 on page 116). Enter the appropriate RPO and schedule information as necessary.
7. Activate the mirror paring and wait for the compare and delta data transfer to complete and the volume to have the RPO_OK status.
8. Proceed with any necessary activities to complete volume resizing on the connected hosts.

4.6 Planning

The most important planning considerations for XIV remote mirroring are those related to ensuring availability and performance of the mirroring connections between XIV systems, and also the performance of the XIV systems. Planning for snapshot capacity usage is also extremely important.

To optimize availability, XIV remote mirroring connections must be spread across multiple ports, on different adapter cards, in different interface modules, and must be connected to different networks. Minimum network bandwidth requirements must be maintained to ensure a stable environment and adequate bandwidth must also be allocated to ensure that the anticipated amount of changed data can be transported across the network between XIV systems within the desired RPO.

Important: Network bandwidths are typically expressed in Megabits per second (Mbps), and disk array bandwidths are expressed in MegaBytes per second (MBps). Although not exact, a factor of 10 between the two will give an acceptable approximation.

To optimize capacity usage, the number and frequency of snapshots (both those required for asynchronous replication and any additional user-initiated snapshots) and the workload change rates must be carefully reviewed. If not enough information is available, a snapshot area that is 30% of the pool size can be used as a starting point. Storage pool snapshot usage thresholds must be set to trigger notification (for example, SNMP, email, SMS) when the snapshot area capacity reaches 50%, and snapshot usage must be monitored continually to understand long-term snapshot capacity requirements.

Important: In asynchronous mirroring, because most-recent and last-replicated snapshots are maintained, it is possible for the snapshot utilization to be three times the rate of change during an interval.

This is because last-replicated snapshots on the master begin as most-recent snapshots and are promoted to last-replicated snapshots, having a lifetime of two intervals, with the new most-recent existing for a single interval, in parallel.

4.7 Advantages of XIV mirroring

XIV remote mirroring provides all the functions that are typical of remote mirroring solutions but, has also the following advantages:

- Both synchronous and asynchronous mirroring are supported on a single XIV system.
- XIV mirroring is supported for consistency groups and individual volumes. Mirrored volumes can be dynamically moved in and out of mirrored consistency groups.
- XIV mirroring is data-aware. Only actual data is replicated.
- Synchronous mirroring automatically resynchronizes couplings when a connection recovers from a network failure.
- Both FC and iSCSI protocols are supported, and both can be used to connect between the same XIV systems. Only one protocol can be active at the time.
- XIV mirroring provides an option to automatically create slave volumes.
- XIV allows user specification of initialization and resynchronization speed.

4.8 Mirroring events

The XIV system generates events, as a result of user actions, components failures, and changes in mirroring status. These events can be used to trigger SNMP traps and send e-mails or text messages.

Thresholds for RPO and for link disruption can be specified by the user and trigger an event when the threshold is reached.

4.9 Mirroring statistics for asynchronous mirroring

The XIV system provides Asynchronous Remote Mirroring performance statistics through both the graphical user interface (GUI) and the command-line interface (XCLI) using the **`mirror_statistics_get`** command.

Performance statistics from the FC or IP network components are also extremely useful for both reporting and troubleshooting activities.

4.10 Boundaries

The XIV Storage System has the following boundaries or limitations:

- Maximum remote systems: The maximum number of remote systems that can be attached to a single primary is 8, with a maximum number of 16 ports on the target
- Number of remote mirrors: The combined number of master and slave volumes (including in mirrored CG) cannot exceed 1536.
- Distance: Distance is limited only by the response time of the medium used. Use asynchronous mirroring when the distance causes unacceptable delays to the host I/O in synchronous mode.

Important: As of XIV Storage V11.2 software, the WAN limitations are a maximum latency of 250 ms, and a minimum constantly available bandwidth of 10 Mbps (static link), or 20 Mbps (dynamic link) for Fibre Channel and 50 Mbps for iSCSI connections is required. The specified minimum bandwidth is a functional minimum and does not necessarily guarantee an acceptable replication speed in a given customer environment and workload.

- Consistency groups are supported within remote mirroring. The maximum number of consistency groups is 256.
- Snapshots: Snapshots are allowed with either the primary or secondary volumes without stopping the mirror. There are also special-purpose snapshots used in the mirroring process. Space must be available in the storage pool for snapshots.
- Master and slave peers cannot be the target of a copy operation and cannot be restored from a snapshot. Peers cannot be deleted or formatted without deleting the coupling first.
- Asynchronous volumes cannot be resized while mirroring is active.

4.11 Using the GUI or XCLI for remote mirroring actions

This section illustrates remote mirroring definition actions through the GUI and XCLI.

4.11.1 Initial setup

When preparing to set up remote mirroring, consider the following questions:

- Will the paths be configured by SAN, FC, or iSCSI?
- Is the port you want to use configured as an initiator or a target?
  - Port 4 default configuration is initiator.
  - Port 2 is suggested as the target port for remote mirror links.
  - Ports can be changed if needed.
- How many pairs will be copied?

  The answer is related to the bandwidth needed between sites.
- How many secondary machines will be used for a single primary?

  Remote mirroring can be set up on paths that are SAN-attached FC or, iSCSI protocols. For most disaster recovery solutions, the secondary system is located at a geographically remote site. The sites will be connected using either SAN connectivity with Fibre Channel Protocol (FCP) or Ethernet with iSCSI.

Important: If the IP network includes firewalls between the mirrored XIV systems, TCP port 3260 must be open within firewalls so that iSCSI replication can work.

Bandwidth considerations must be taken into account when planning the infrastructure to support the remote mirroring implementation. Knowing when the peak write rate occurs for systems attached to the storage will help with the planning for the number of paths needed to support the remote mirroring function and any future growth plans.

When the protocol has been selected, it is time to determine which ports on the XIV Storage System will be used. The port settings are easily displayed using the XCLI session environment and the command **`fc_port_list`** for Fibre Channel or **`ipinterface_list`** for iSCSI.

There must always be a minimum of two paths configured within remote mirroring for FCP connections, and these paths must be dedicated to remote mirroring. These two paths must be considered a set. Use port 4 and port 2 in the selected interface module for this purpose. For redundancy, additional sets of paths must be configured in different interface modules.

Fibre Channel paths for remote mirroring have slightly more requirements for setup, which is a method that is explored here first.

As Example 4-1 shows in the Role column, each Fibre Channel port is identified as a target or an initiator. Simply put, a target in a remote mirror configuration is the port that will be receiving data from the other system, whereas an initiator is the port that will be doing the sending of data. In this example, there are three initiators configured. Initiators, by default, are configured on FC:*X*:4 (*X* is the module number). In this example, port 4 on all six interface modules is configured as the initiator.

Example 4-1 The fc_port_list output command

```
>> fc_port_list
Component ID   Status   Currently Functioning   WWPN               Port ID   Role
1:FC_Port:4:1  OK       yes                     5001738000130140   00030A00  Target
1:FC_Port:4:2  OK       yes                     5001738000130141   0075002E  Target
1:FC_Port:4:3  OK       yes                     5001738000130142   00750029  Target
1:FC_Port:4:4  OK       yes                     5001738000130143   00750027  Initiator
1:FC_Port:5:1  OK       yes                     5001738000130150   00611000  Target
1:FC_Port:5:2  OK       yes                     5001738000130151   0075001F  Target
1:FC_Port:5:3  OK       yes                     5001738000130152   00021D00  Target
1:FC_Port:5:4  OK       yes                     5001738000130153   00000000  Initiator
1:FC_Port:6:1  OK       yes                     5001738000130160   00070A00  Target
1:FC_Port:6:2  OK       yes                     5001738000130161   006D0713  Target
1:FC_Port:6:3  OK       yes                     5001738000130162   00000000  Target
1:FC_Port:6:4  OK       yes                     5001738000130163   0075002F  Initiator
1:FC_Port:9:1  OK       yes                     5001738000130190   00DDEE02  Target
1:FC_Port:9:2  OK       yes                     5001738000130191   00FFFFFF  Target
1:FC_Port:9:3  OK       yes                     5001738000130192   00021700  Target
1:FC_Port:9:4  OK       yes                     5001738000130193   00021600  Initiator
1:FC_Port:8:1  OK       yes                     5001738000130180   00060219  Target
1:FC_Port:8:2  OK       yes                     5001738000130181   00021C00  Target
1:FC_Port:8:3  OK       yes                     5001738000130182   002D0027  Target
1:FC_Port:8:4  OK       yes                     5001738000130183   002D0026  Initiator
1:FC_Port:7:1  OK       yes                     5001738000130170   006B0F00  Target
1:FC_Port:7:2  OK       yes                     5001738000130171   00681813  Target
1:FC_Port:7:3  OK       yes                     5001738000130172   00021F00  Target
1:FC_Port:7:4  OK       yes                     5001738000130173   00021E00  Initiator
```

The iSCSI connections that are shown in Example 4-2 use the **ipinterface_list** command. The output is truncated to show only the iSCSI connections that are of interest here. The command also displays all Ethernet connections and settings. In this example, two connections are displayed for iSCSI: one connection in module 7 and one in module 8.

Example 4-2 The ipinterface_list command

```
>> ipinterface_list
Name         Type   IP Address    Network Mask   Default Gateway  MTU   Module       Ports
itso_m8_p1   iSCSI  9.11.237.156 255.255.254.0  9.11.236.1       4500  1:Module:8   1
itso_m7_p1   iSCSI  9.11.237.155 255.255.254.0  9.11.236.1       4500  1:Module:7   1
```

Alternately, one can query for the existing connectivity among the managed XIVs, by selecting a system in the GUI, followed by selecting **XIV Connectivity** (Figure 4-44).

Figure 4-44 Selecting mirror connectivity

Click the connecting links between the systems of interest to view the ports.

Right-click a specific port and select **Properties**; the output is shown in Figure 4-45 on page 102. This particular port is configured as a target, as indicated in the Role field.

Figure 4-45 Port properties displayed with GUI

Another way to query the port configuration is to select the desired system, click the curved arrow (at the bottom right of the window) to display the ports on the back of the system, and hover the mouse over a port, as shown in Figure 4-46. This view displays all the information that is shown in Figure 4-45.

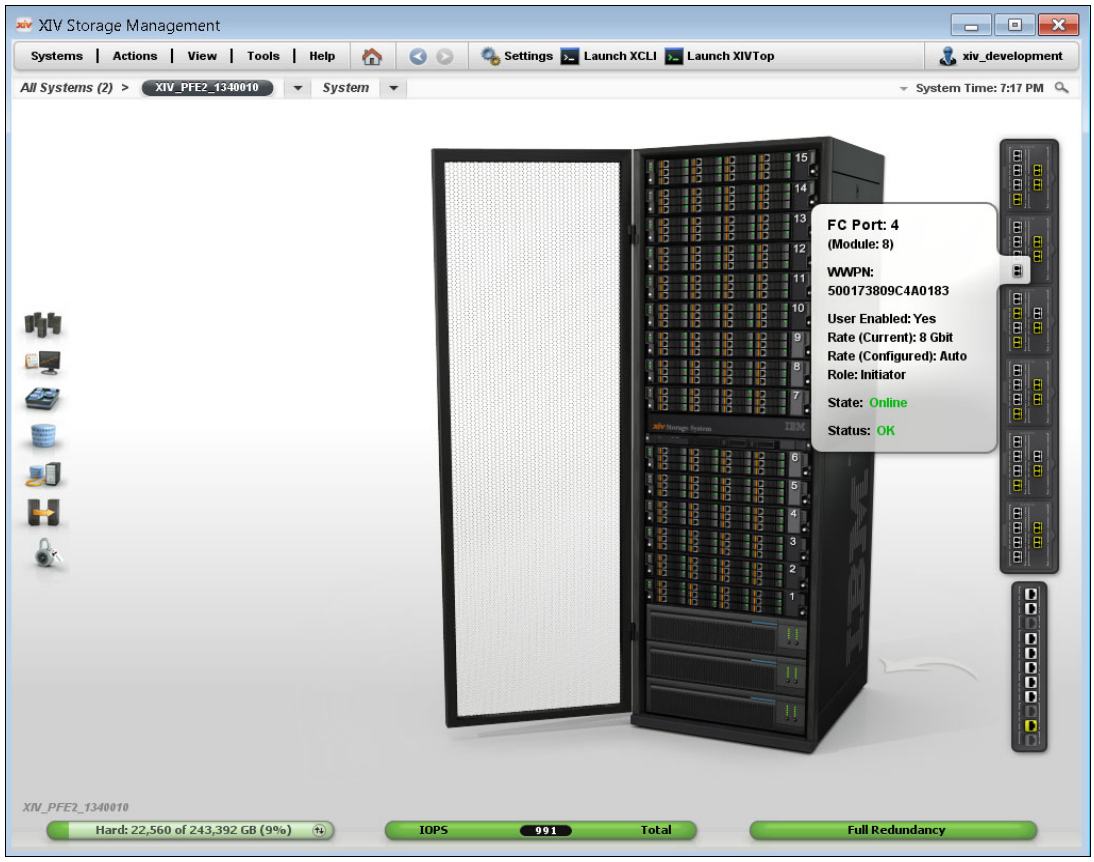


Figure 4-46 Port information from the patch panel view

Similar information can be displayed for the iSCSI connections using the GUI, as shown in Figure 4-47. This view can be seen either by right-clicking the Ethernet port (similar to the Fibre Channel port shown in Figure 4-47) or by selecting the system, then selecting **Hosts and LUNs → iSCSI Connectivity**. This sequence displays the same two iSCSI definitions that are shown with the XCLI command.

Name	Address	Netmask	Gateway
iSCSI_M5_P1	9.155.115.180	255.255.240.0	9.155.112.1
iSCSI_M7_P1	9.155.115.181	255.255.240.0	9.155.112.1

Figure 4-47 iSCSI connectivity

By default, Fibre Channel ports 2 and 4 (target and initiator, respectively) from every module are designed to be used for remote mirroring. For example, port 4 module 8 (initiator) on the local machine is connected to port 2 module 8 (target) on the remote machine. When setting up a new system, it is best to plan for any remote mirroring and reserve these ports for that purpose. However different ports could be used as needed.

If a port role does need changing, you can change the port role with both the XCLI and the GUI. Use the XCLI **`fc_port_config`** command to change a port, as shown in Example 4-3. Using the output from **`fc_port_list`**, you can get the fc_port name to be used in the command, changing the port role to be either initiator or target, as needed.

Example 4-3 XCLI command to configure a port

```
fc_port_config fc_port=1:FC_Port:4:3 role=initiator
Command completed successfully

fc_port_list
Component ID  Status  Currently Functioning  WWPN              Port ID  Role
1:FC_Port:4:3 OK      yes                    5001738000130142 00750029 Initiator
```

To perform the same function with the GUI, select the primary system, open the patch panel view, and right-click the port, as shown in Figure 4-48.

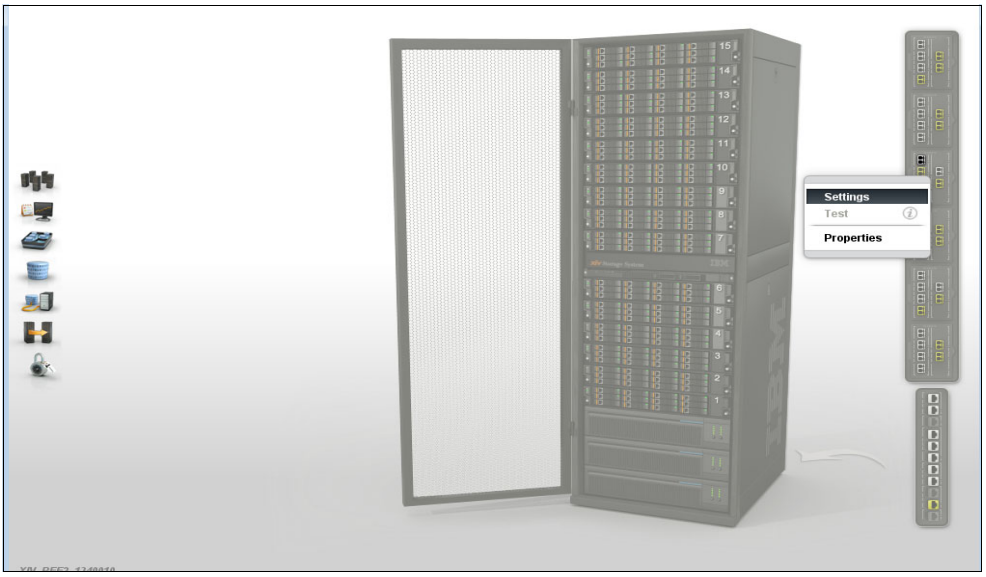


Figure 4-48 XIV Main window: Right-click the connection panel to configure ports

Selecting **Settings** opens a configuration window, as shown in Figure 4-49, which allows the port to be enabled (or disabled), its role defined as target or initiator, and, finally, the speed for the port configured (Auto, 1, 2, 4 Gbps; additionally on Gen3, 8 Gbps is available).

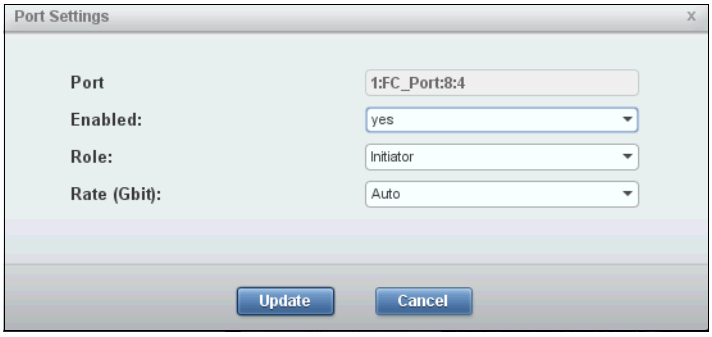


Figure 4-49 Configure port with GUI

Planning for remote mirroring is important when determining how many copy pairs will exist. All volumes defined in the system can be mirrored. A single primary system is limited to a maximum of 8 secondary systems. Volumes cannot be part of an XIV data migration and a remote mirror volume at the same time. Data migration information is in Chapter 9, “Data migration” on page 245.

4.11.2 Remote mirror target configuration

The connections to the target (secondary) XIV system must be defined. The assumption here is that the physical connections and zoning were set up. Target configuration is done from the mirror connectivity menu using the following steps:

1. Add the target system by right-clicking the system image and selecting **Create Target**, as shown in Figure 4-50.

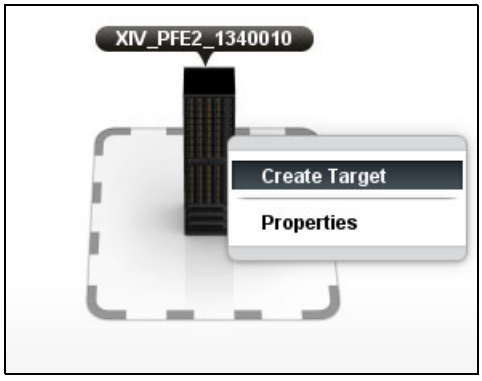


Figure 4-50 Create target

2. Define the type of mirroring to be used (mirroring or migration) and the type of connection (iSCSI or FC), as shown in Figure 4-51.

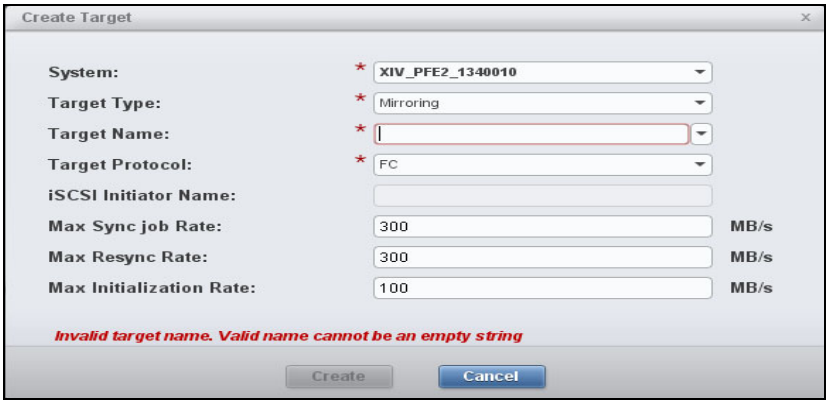


Figure 4-51 Target type and protocol

3. As shown in Figure 4-52, define connections by clicking the line between the two XIV systems to display the link status detail window.

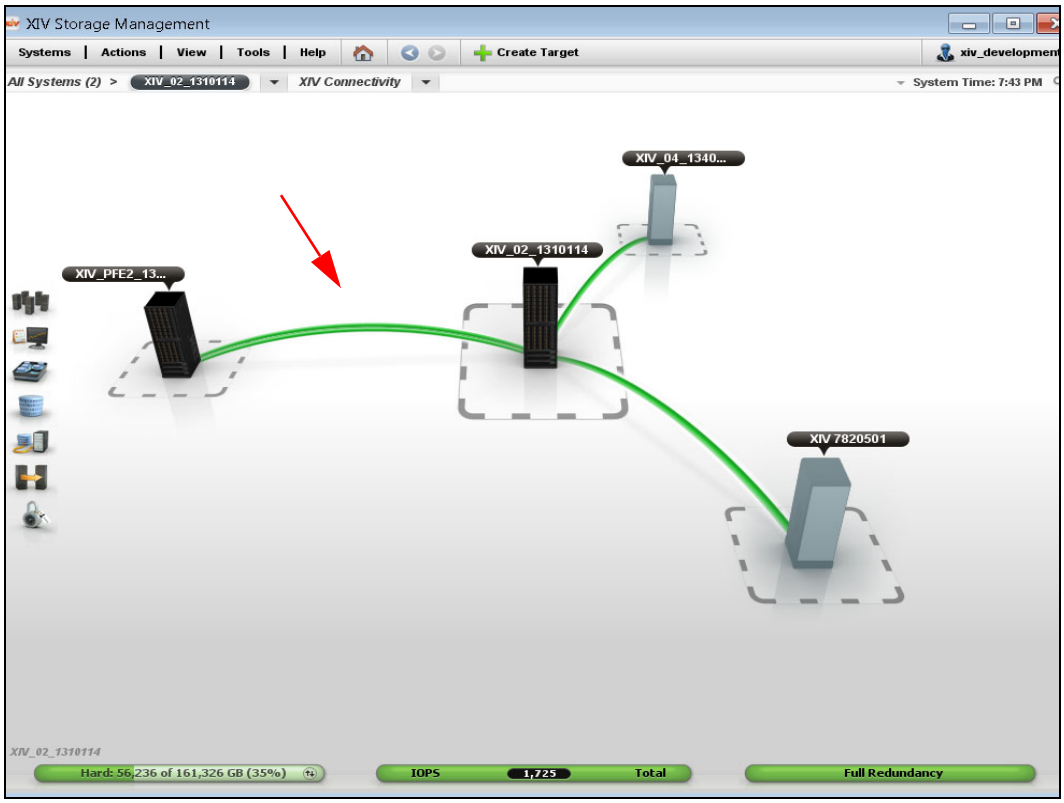


Figure 4-52 Define and Update XIV connectivity panel; three established links shown

Connections are easily defined by clicking **Show Auto Detected Connections**. This shows the possible connections and provides an Approve button to define the detected connections. Remember that for FCP ports an initiator must be connected to a target and the proper zoning must be established for the connections to be successful.

The possible connections are shown in green, as depicted in Figure 4-53.

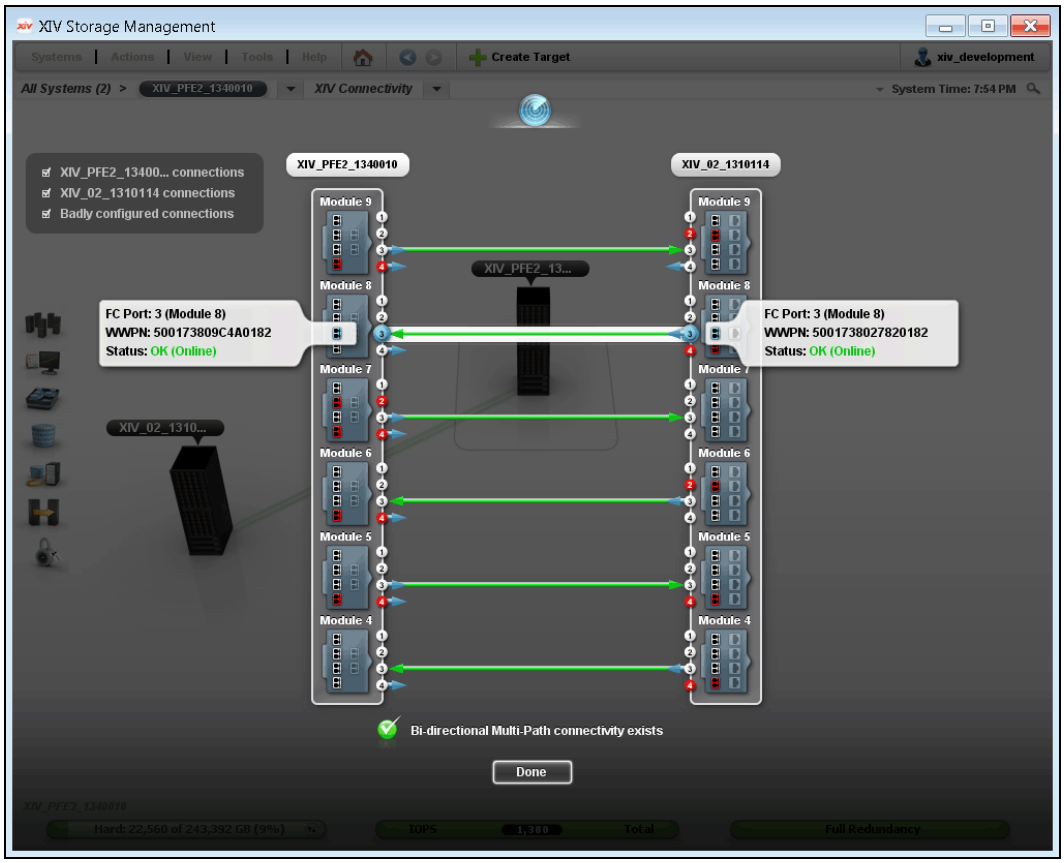


Figure 4-53 Mirroring Connections, with mouse highlight showing WWPNs

4. Connections can also be defined by clicking a port on the primary system and dragging the corresponding port on the target system. This is shown as a blue line in Figure 4-54.

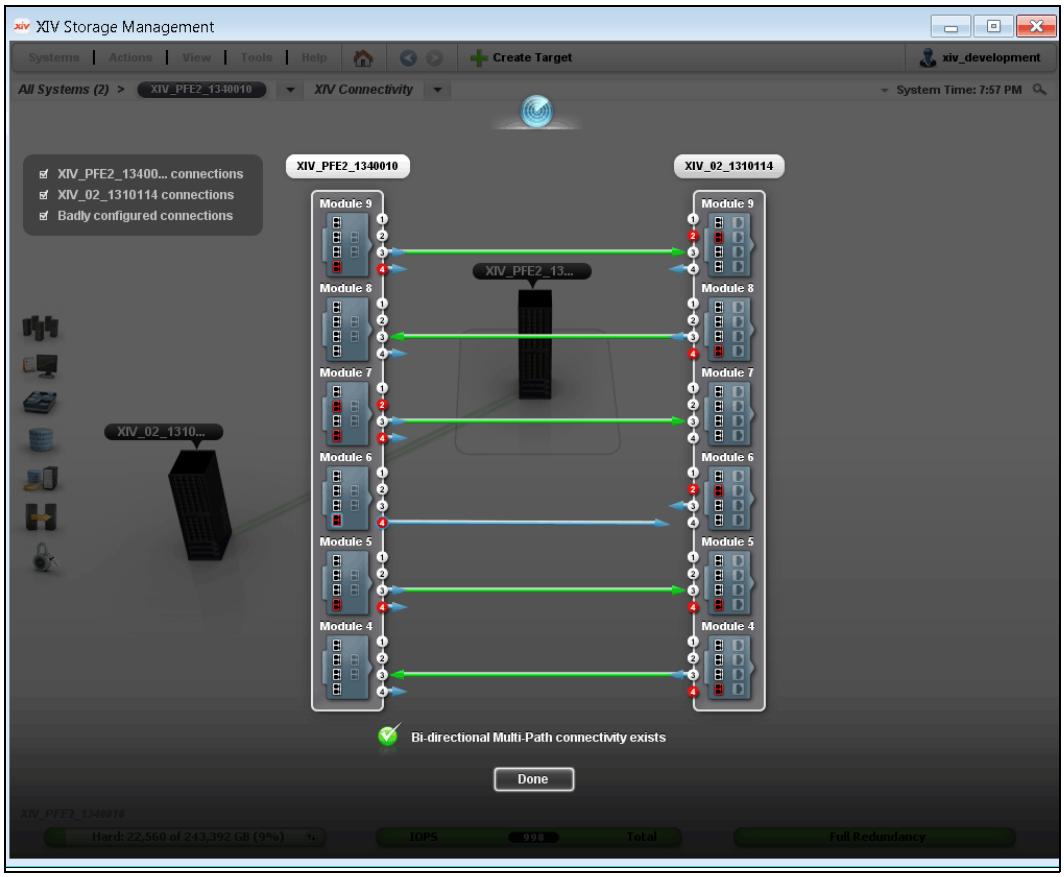


Figure 4-54 Creating a connection; mouse pointer drags link to port 4; module 6 on the right side

5. Release the mouse button to initiate the connection; then the status can be displayed, as shown in Figure 4-55.

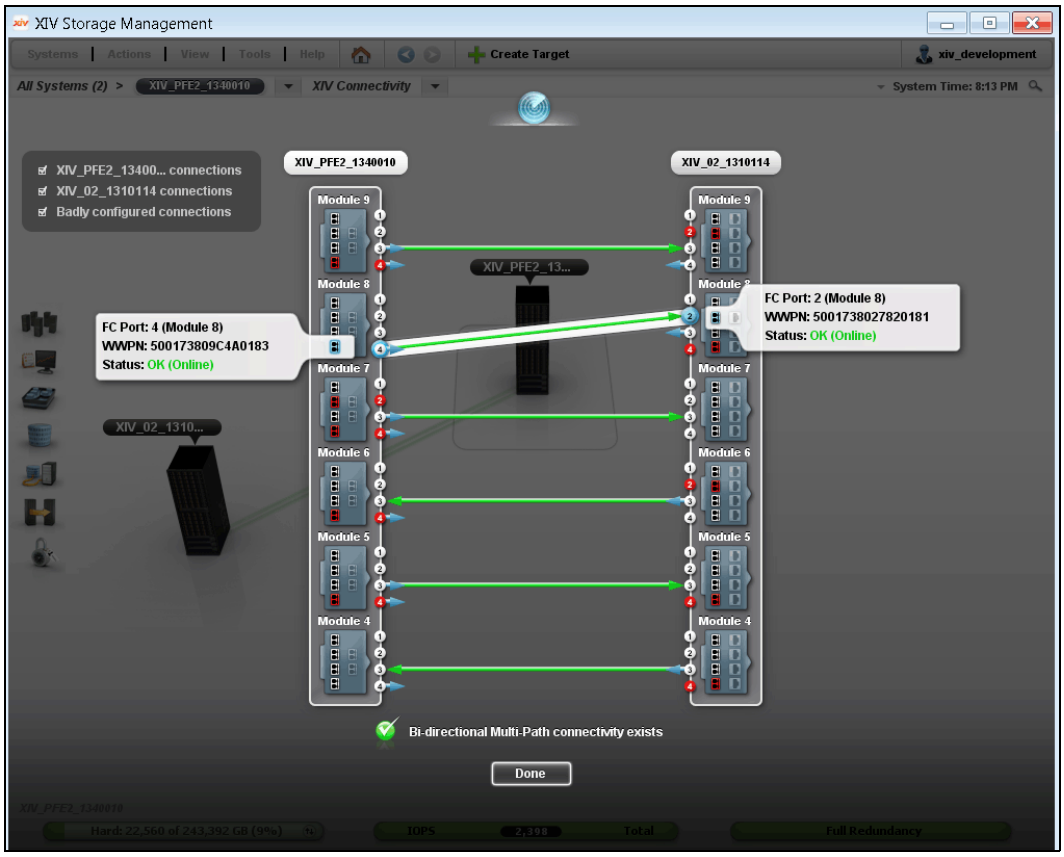


Figure 4-55 Define connection and view status completed; both links green

6. Right-click a path and you have options to Activate, Deactivate, and Delete the selected path, as shown in Figure 4-56.

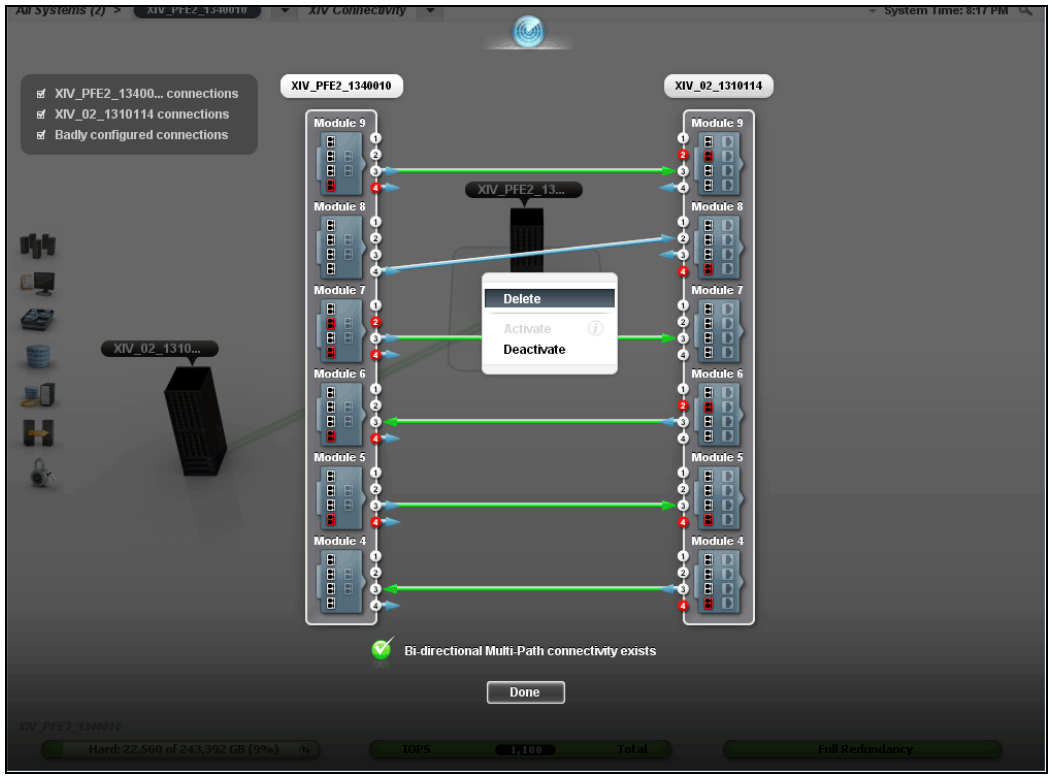


Figure 4-56 Paths actions menu; blue link is highlighted

To delete the connections between two XIV systems, complete the following steps:

1. Delete all paths between the two systems.
2. In the Mirroring Connectivity display, delete the target system, as shown in Figure 4-57.

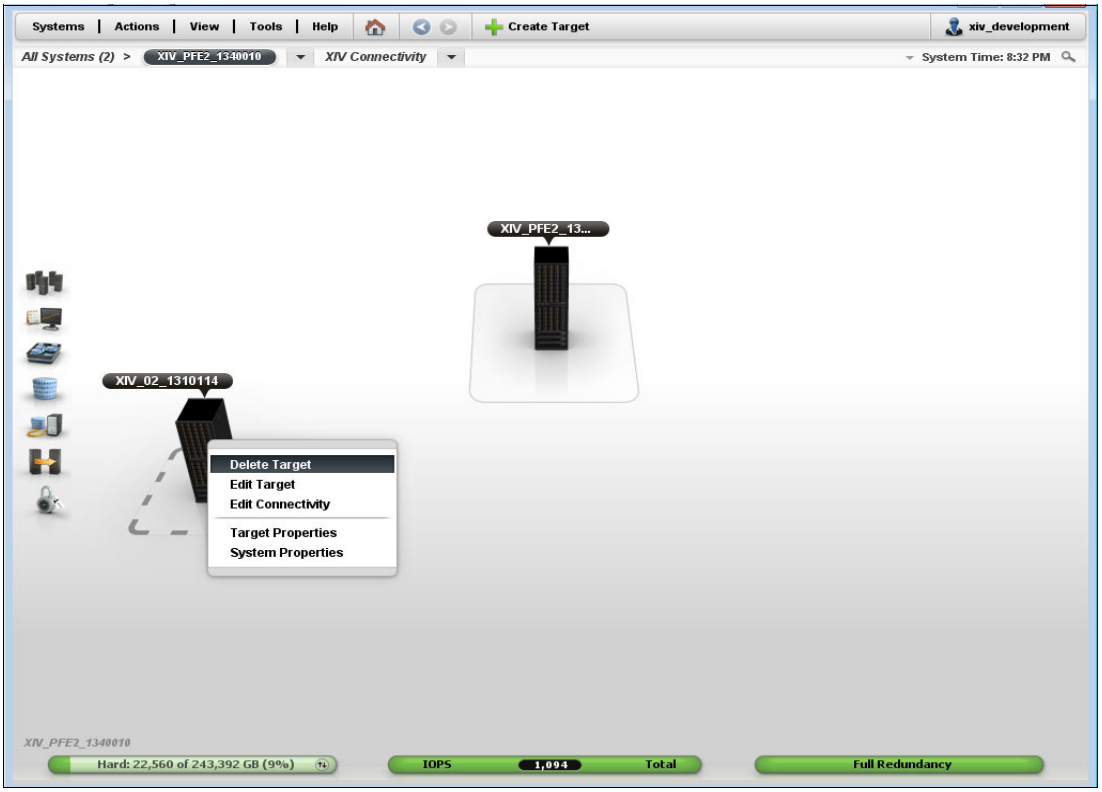


Figure 4-57 Delete Target XIV; note links have already been removed

4.11.3 XCLI examples

XCLI commands can be used to configure connectivity between the primary XIV system and the target or secondary XIV system (Figure 4-58).

```
target_define target="WSC_1300331"  protocol=FC xiv_features=yes
target_mirroring_allow target="WSC_1300331"
target_define target="WSC_6000639" system_id=639  protocol=FC xiv_features=yes
target_mirroring_allow target="WSC_6000639"

target_port_add fcaddress=50017380014B0183 target="WSC_1300331"
target_port_add fcaddress=50017380027F0180 target="WSC_6000639"
target_port_add fcaddress=50017380014B0193 target="WSC_1300331"
target_port_add fcaddress=50017380027F0190 target="WSC_6000639"
target_port_add fcaddress=50017380027F0183 target="WSC_6000639"
target_port_add fcaddress=50017380014B0181 target="WSC_1300331"
target_connectivity_define local_port="1:FC_Port:8:4"
   fcaddress=50017380014B0181 target="WSC_1300331"
target_port_add fcaddress=50017380027F0193 target="WSC_6000639"
target_port_add fcaddress=50017380014B0191 target="WSC_1300331"
target_connectivity_define local_port="1:FC_Port:9:4"
   fcaddress=50017380014B0191 target="WSC_1300331"
target_connectivity_define target="WSC_6000639" local_port="1:FC_Port:8:4"
   fcaddress="50017380027F0180"
target_connectivity_define target="WSC_6000639" local_port="1:FC_Port:9:4"
   fcaddress="50017380027F0190"
```

Figure 4-58 Define target XCLI commands

XCLI commands can also be used to delete the connectivity between the primary XIV System and the secondary XIV system (Figure 4-59).

```
target_connectivity_delete local_port="1:FC_Port:8:4"
   fcaddress=50017380014B0181 target="WSC_1300331"
target_port_delete fcaddress=50017380014B0181 target="WSC_1300331"
target_connectivity_delete local_port="1:FC_Port:8:4"
   fcaddress=50017380027F0180 target="WSC_6000639"
target_port_delete fcaddress=50017380027F0180 target="WSC_6000639"
target_connectivity_delete local_port="1:FC_Port:9:4"
   fcaddress=50017380014B0191 target="WSC_1300331"
target_port_delete fcaddress=50017380014B0191 target="WSC_1300331"
target_connectivity_delete local_port="1:FC_Port:9:4"
   fcaddress=50017380027F0190 target="WSC_6000639"

target_port_delete fcaddress=50017380027F0190 target="WSC_6000639"
target_port_delete target="WSC_6000639" fcaddress="50017380027F0183"
target_port_delete target="WSC_6000639" fcaddress="50017380027F0193"
target_delete target="WSC_6000639"
target_port_delete target="WSC_1300331" fcaddress="50017380014B0183"
target_port_delete target="WSC_1300331" fcaddress="50017380014B0193"
target_delete target="WSC_1300331"
```

Figure 4-59 Delete target XCLI commands

4.12 Configuring remote mirroring

Configuration tasks differ depending on the nature of the coupling. Synchronous and asynchronous mirroring are the two types of coupling supported. See the following chapters for more information:

- For specific configuration tasks related to synchronous mirroring, see Chapter 5, “Synchronous Remote Mirroring” on page 113.
- For specific configuration tasks related to asynchronous mirroring, see Chapter 6, “Asynchronous remote mirroring” on page 149.

5

Synchronous Remote Mirroring

Synchronous remote mirroring is a methodology for data replication between two storage systems which achieves a recovery point objective (RPO) of zero. This implies that write operations from the hosts are not acknowledged by the storage device before the data has been written successfully to both, local and remote, systems. The purpose is to have a copy of important data available in the case a disaster happens on one of the two sites.

This chapter describes synchronous remote mirroring and available options. Topics covered include configuration, usage, and disaster recovery considerations.

© Copyright IBM Corp. 2014. All rights reserved.

5.1 Synchronous mirroring considerations

A mirror relationship or coupling, consists of a primary and a secondary site. The primary site is usually designated as the main site that serves the active hosts. A secondary site, which holds a backup of the data, is used if the primary site is unavailable because of, for example. a complete power outage, a fire, or some other disaster.

Besides the designation as primary or secondary site, a role is assigned, which can be either master or slave. The site that has the master role acts as the source for the replication; the site that has the slave role acts as the destination for the replication.

In a normal operational mode, the primary site also holds the master role and the secondary site holds the slave role. Those roles can be changed by using the **`mirror_change_role`** and **`mirror_switch_roles`** commands or by the respective GUI function.

Synchronous mirroring provides continuous availability of information by ensuring that a secondary site maintains the same consistent data as the primary site with a zero recovery point objective (RPO).To accomplish this, at all times a write operation will be acknowledged to the host only if it was successfully written on both storage systems.

Note: If the link between the two storage systems stops working, hosts will continue to receive acknowledgements from the primary device thereby preventing an access loss. When the link is working again, any unreplicated data is automatically be resynced to the remote site.

The remote mirroring configuration process involves configuring volumes or consistency groups (CGs) into pairs. When a pair of volumes or CGs enters into a remote mirror relationship and is managed as such, it is referred to as a *coupling*. In this chapter, the assumption is that links between the primary and secondary XIV are already established, as described in 4.11.2, “Remote mirror target configuration” on page 104.

Important:

- For mirroring, a reliable, dedicated network link is preferred. Links can be shared, but require an available and consistent network bandwidth. The minimum specified bandwidth is 10 Mbps for FC and 20 Mbps for iSCSI (IBM XIV Software V11.1.x or later). This is a functional minimum and does not guarantee a reliable replication path. Moreover the customer network environment and the I/O workload must be taken into account to ensure a steady connectivity between the two sites.
- The required minimum bandwidth should *not* be based on a time-averaged calculation, as typically reported by network monitoring tools. Instead, the maximum instantaneous workload should be the base for the bandwidth required. Typically this can be achieved by using network quality of service (QoS) control.
- Although not shown in most diagrams, a switch is required to connect the IBM XIV systems being mirrored, which means a direct connection is not supported.
- Typically the distance between two sites does not exceed 100 km because of transmission latency. Beyond that distance, consider asynchronous replication.

Other considerations with mirrored volumes

Consider the following information:

- Renaming a volume changes the name of the last-consistent and most-updated snapshots.
- Deleting all snapshots does not delete the last-consistent and most-updated snapshot.
- Resizing a primary volume resizes its secondary volume.
- A primary volume cannot be resized when the link is down.
- Resizing, deleting, and formatting are not permitted on a secondary volume.
- A primary volume cannot be formatted. If a primary volume must be formatted, an administrator must first deactivate the mirroring, delete the mirroring, format both the secondary and primary volumes, and then define the mirroring again.
- Secondary or primary volumes cannot be the target of a copy operation.
- Locking and unlocking are not permitted on a secondary volume.
- The last-consistent and most-updated snapshots cannot be unlocked.
- Deleting is not permitted on a primary volume.
- Restoring from a snapshot is not permitted on a primary volume.
- Restoring from a snapshot is not permitted on a secondary volume.
- A snapshot cannot be created with the same name as the last-consistent or most-updated snapshot.

5.2 Setting up mirroring

To set up mirroring, use the XIV Storage System GUI or the XCLI.

5.2.1 Using the GUI for volume mirroring setup

In the GUI, select the primary IBM XIV and choose **Remote** → **Mirroring**, as shown in Figure 5-1.

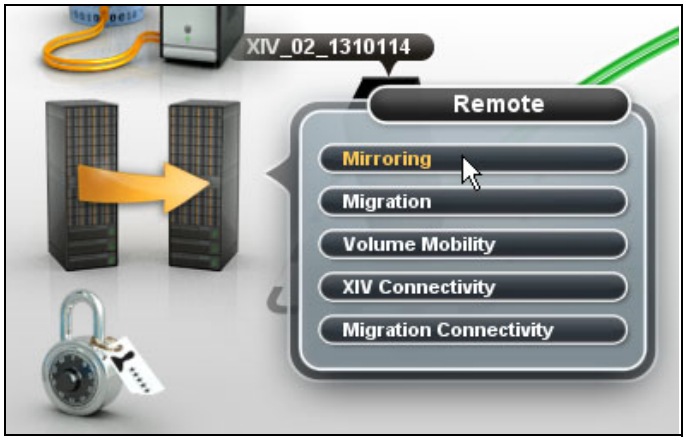


Figure 5-1 Selecting Mirroring

To create a mirror, complete the following steps:

1. Select **Mirror Volume/CG** (Figure 5-2) and specify the master system for the mirror pair

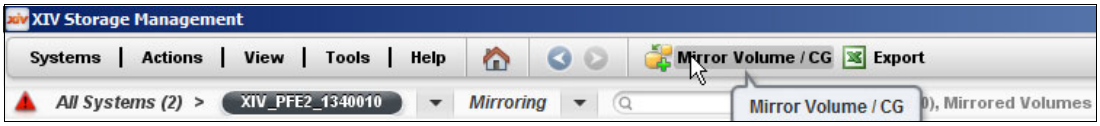


Figure 5-2 Create Mirror Volume/CG

There are other ways to create a mirror pair from within the GUI. For example, in the Volumes and Snapshots panel you can right-click a volume and then select **Mirror Volume** from there.

Note: It is not possible to mirror a snapshot volume.

The Create Mirror dialog box opens (Figure 5-3).

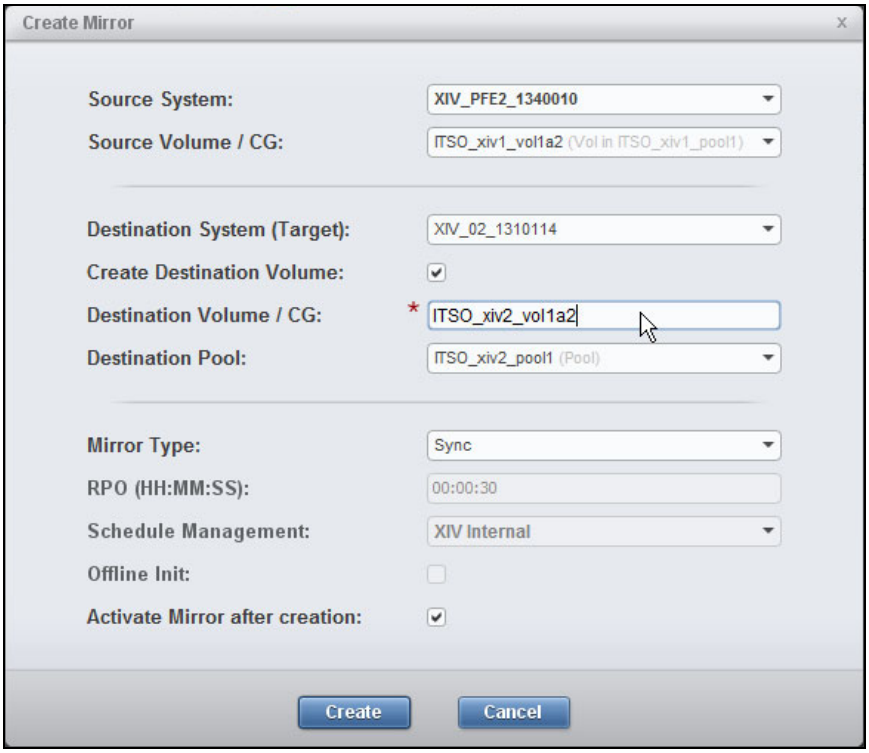


Figure 5-3 Create Mirror Input panel

2. Make the appropriate selections and entries:
   - Source System

     This is the IBM XIV at the primary site that will contain the source volume or CG. Select the primary system from a list of known sources.
   - Source Volume / CG

     This is the volume/CG at the primary site to be mirrored. Select the volume/CG from the list. The consistency groups are shown at the bottom of the list, per pool.

– Destination System (Target)

  This is the IBM XIV at the secondary site that will contain the target volume or CG. Select the secondary system from a list of known targets.

– Create Destination Volume

  If selected, a destination volume will be created automatically in the selected destination pool. If unselected, you must specify the volume manually. By default, the volume name and size are taken from the source volume.

– Destination Volume / CG

  This is the name of the destination volume or CG. If the **Create Destination Volume** option was selected, a destination volume of same size and name as the source will be automatically created in the chosen pool. The name can be changed. If the **Create Destination Volume** option was not selected, the target volume or CG must exist and can be selected from the list. It must have exactly the same size as the source volume and needs to be formatted, except if ***Offline Init*** (see below) is chosen.

– Destination Pool

  This is the Storage pool on the secondary IBM XIV that will contain the destination volume. The pool must already exist; this option is available only if **Create Destination Volume** is selected.

– Mirror Type

  Select **Sync** for synchronous mirroring. *Async* is for asynchronous replication, which is described in Chapter 6, “Asynchronous remote mirroring” on page 149.

– RPO (HH:MM:SS)

  This option is disabled if Mirror Type is Sync. RPO stands for recovery point objective and is only relevant for asynchronous mirroring. By definition, synchronous mirroring has a zero RPO.

– Schedule Management

  This option is disabled if Mirror Type is Sync. Schedule Management is relevant only for asynchronous mirroring.

– Offline Init

  Select this offline initialization option (also known as *truck* mode) if the source volume or CG has already been copied to the target system by some other means, for example by transferring the data manually by tape or hard disk. Upon activation of the mirror, only the differences between the source and the destination volume/CG need to be transmitted across the mirror link. This option is useful if the amount of data is huge and synchronization might take more time, because of the available bandwidth, than a manual transport.

  Note: Offline init for synchronous mirroring is introduced in release 11.4. It also facilitates to switch from asynchronous to synchronous replication, which was not possible before.

– Activate Mirror after creation

  This option activates the mirror immediately after its creation, and thereby reduces the number of clicks (versus doing it after on the Mirror Volume / CG menu).

3. After all the appropriate entries have been completed, click **Create**.

   A coupling is created and is in standby, inactive mode, as shown in Figure 5-4. In this state data is not yet copied from the source to the target volume.

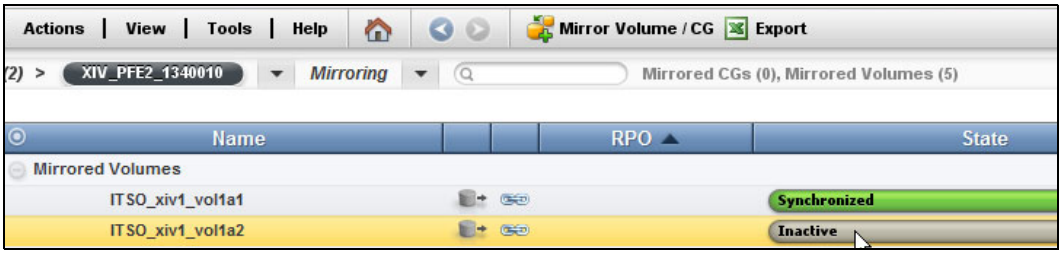


Figure 5-4 Coupling on the primary IBM XIV in standby, inactive mode

A corresponding coupling is automatically created on the secondary XIV, and it is also in standby, inactive mode as seen in Figure 5-5.

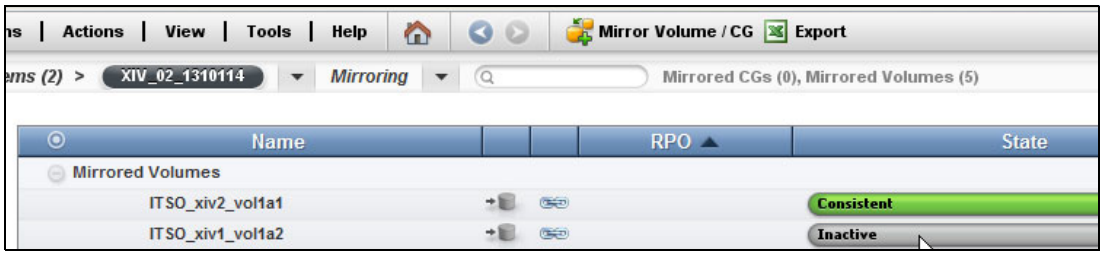


Figure 5-5 Coupling on the secondary IBM XIV in standby, inactive mode

4. Repeat steps 1 - 3 to create additional couplings.

5.2.2 Using the GUI for volume mirror activation

To activate the mirror, proceed as follows:

1. On the primary IBM XIV, go to **Remote** → **Mirroring.** Highlight all the couplings that you want to activate, right-click, and select **Activate**, as shown in Figure 5-6.

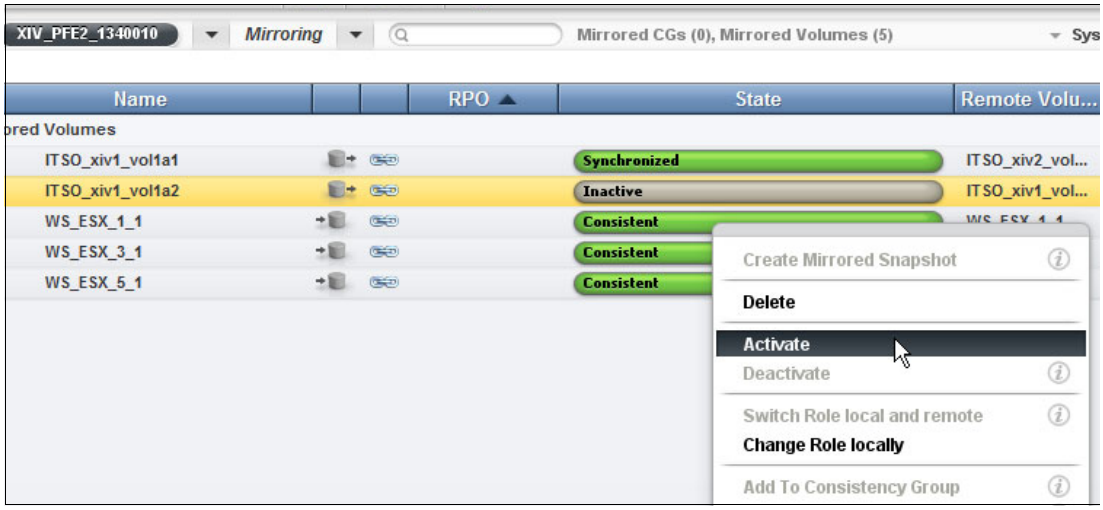


Figure 5-6 Mirror activation

2. The mirror then enters an *Initialization* state on both primary and secondary IBM XIVs as shown in Figure 5-7 and Figure 5-8.

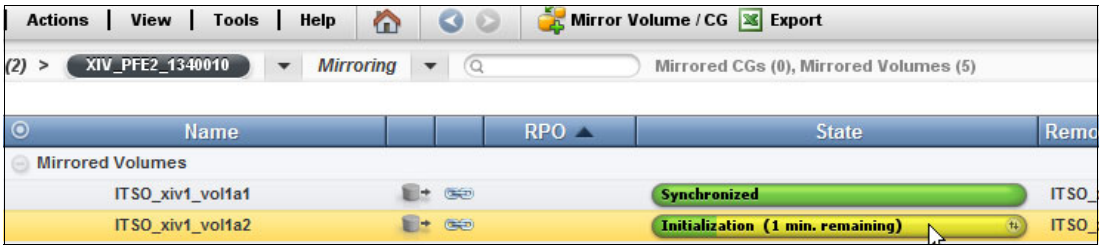


Figure 5-7 Mirror initialization phase (primary)

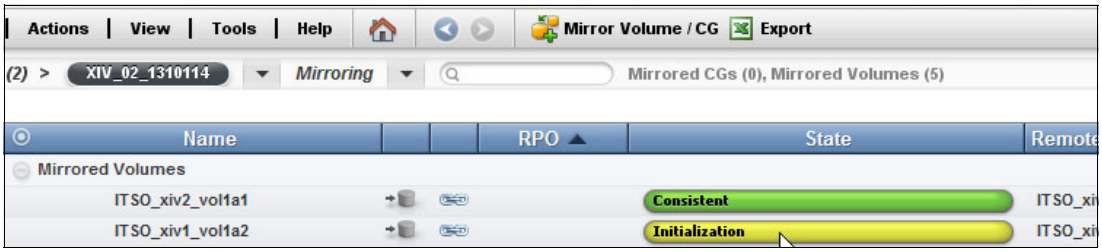


Figure 5-8 Mirror initialization phase (secondary)

3. After the *initialization* phase is complete, the primary's state is *synchronized* and the secondary's state is *Consistent* as shown in Figure 5-9 and Figure 5-10.

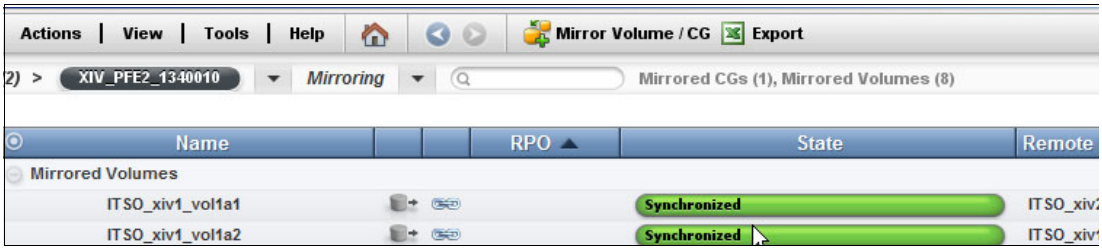


Figure 5-9 Mirror synchronized (primary)

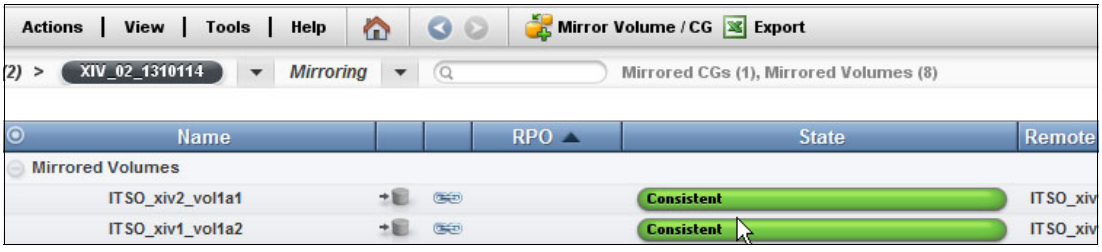


Figure 5-10 Mirror consistent (secondary)

4. Repeat activating all remaining volume mirrors until all are in the *Initialization* state. Getting to synchronized state (for primary volume) or consistent state (for secondary volume) might take some time depending on the amount of data to be transferred and the bandwidth of the link.

Note: After mirroring is active, resizing the source volume will automatically resize the target volume to match.

5.2.3 Using XCLI for volume mirroring setup

Tip: When working with the XCLI session or the XCLI from a command prompt, the storage system to which the XCLI commands are directed is not necessarily visible. Commands can inadvertently be run against the wrong IBM XIV. Therefore, a good idea is to issue a **config_get** command to verify that the intended IBM XIV is being addressed.

To set up volume mirroring using XCLI, follow these steps:

1. Open an XCLI session for the primary IBM XIV and run the **mirror_create** command shown in Example 5-1.

Example 5-1 Create remote mirror coupling

```
XIV_PFE2_1340010>>mirror_create target="XIV_02_1310114" vol="ITSO_xiv1_vol1a2"
slave_vol="ITSO_xiv2_vol1a2" remote_pool="ITSO_xiv2_pool1" create_slave=yes
Command executed successfully.
```

2. To list the couplings on the primary IBM XIV, run the **mirror_list** command shown in Example 5-2. Note that the *Initializing* status is used when the coupling is in standby, inactive, or initializing state.

Example 5-2 Listing mirror couplings on the primary

```
XIV_PFE2_1340010>>mirror_list
Name              Mirror Type       Mirror Object  Role   Remote System   Remote Peer       Active  Status       Link Up
ITSO_xiv1_vol1a1  sync_best_effort  Volume        Master  XIV_02_1310114  ITSO_xiv2_vol1a1  yes     Synchronized yes
ITSO_xiv1_vol1a2  sync_best_effort  Volume       Master  XIV_02_1310114  ITSO_xiv2_vol1a2  no      Initializing yes
```

3. To list the couplings on the secondary IBM XIV, run the **mirror_list** command shown in Example 3. Note the status of *Initializing* is used when the coupling is in a standby, inactive, or initializing state.

Example 5-3 Listing mirror couplings on the secondary

```
XIV_02_1310114>>mirror_list
Name              Mirror Type       Mirror Object  Role   Remote System     Remote Peer       Active Status       LinkUp
ITSO_xiv2_vol1a1  sync_best_effort  Volume         Slave  XIV_PFE2_1340010  ITSO_xiv1_vol1a1  yes    Consistent   yes
ITSO_xiv2_vol1a2  sync_best_effort  Volume         Slave  XIV_PFE2_1340010  ITSO_xiv1_vol1a2  no     Initializing yes
```

4. Repeat steps 1 - 3 to create additional mirror couplings.

5.2.4 Using XCLI for volume mirror activation

To activate the mirror, proceed as follows:

1. On the primary XIV, run the **mirror_activate** command shown in Example 5-4.

Example 5-4 Activating the mirror coupling

```
XIV_PFE2_1340010>>mirror_activate vol=ITSO_xiv1_vol1a2
Command executed successfully.
```

2. On the primary XIV, run the **mirror_list** command (Example 5-5 on page 121) to see the status of the couplings.

Example 5-5 List remote mirror status on the primary XIV

```
XIV_PFE2_1340010>>mirror_list
Name                    Mirror Type          Mirror Object Role     Remote System   Remote Peer           Active Status
        Link Up
ITSO_xiv1_vol1a1   sync_best_effort   Volume          Master XIV_02_1310114  ITSO_xiv2_vol1a1   yes      Synchronized   yes
ITSO_xiv1_vol1a2   sync_best_effort   Volume          Master XIV_02_1310114  ITSO_xiv2_vol1a2   yes      Synchronized   yes
```

3. On the secondary XIV, run the **mirror_list** command (Example 5-6) to see the status of the couplings.

Example 5-6 List remote mirror status on the secondary XIV

```
XIV_02_1310114>>mirror_list
Name                    Mirror Type          Mirror Object  Role   Remote System          Remote Peer           Active   Status
Link Up
ITSO_xiv2_vol1a1   sync_best_effort   Volume          Slave  XIV_PFE2_1340010   ITSO_xiv1_vol1a1   yes       Consistent      yes
ITSO_xiv2_vol1a2   sync_best_effort   Volume          Slave  XIV_PFE2_1340010   ITSO_xiv1_vol1a2   yes       Initializing    yes
```

4. Repeat steps 1 - 3 to activate additional couplings.

5.3 Setting up mirroring for a consistency group

A consistency group is an administrative unit of multiple volumes and facilitates simultaneous snapshots of multiple volumes, mirroring of volume groups, and administration of volume sets. Setting a consistency group to be mirrored requires these steps:

1. Creating consistency groups on both sites.
2. Setting them to be mirrored.
3. Populating the CG on the primary site with mirrored volumes.

A consistency group must be created on the primary IBM XIV and then a corresponding consistency group must be created on the secondary IBM XIV.

5.3.1 Considerations regarding a consistency group

The following considerations apply:

- Volumes and CG must be on the same XIV system.
- Volumes and CG must belong to the same pool.
- A volume can belong to only one CG.
- A CG must be mirrored at first, thereafter volumes can be added.
- Each volume to be added must be mirrored itself.
- Each volume to be added must be in the same mirroring state.
- Mirror operations are not allowed during initialization of a volume/CG.
- Target pool and consistency group have to be defined on the slave XIV.
- The following volume mirroring settings must be identical to those of the CG:
  - Mirroring type
  - Mirroring role
  - Mirroring status
  - Mirroring target
  - Target pool
  - Target CG

- It is possible to add a mirrored volume to a non-mirrored consistency group and have the volume maintain its mirroring settings. A mirror for the CG cannot be set up thereafter.
- Removing a mirrored volume from a mirrored CG also removes the peer volume from the slave CG. Volume mirroring thereafter will carry on with the same settings. It is also possible to add the volume back to the CG where it was removed from.
- The Create Snapshot Group function allows you to create a set of snapshots of all volumes of a CG with the same time stamp. It does not affect the mirroring operation of the CG.

5.3.2 Using the GUI for CG mirroring setup

To create a mirrored consistency group, use the following steps:

1. Select **Volumes** → **Consistency Groups** (Figure 5-11).

Figure 5-11 Consistency Groups

2. If a CG already has a volume, assigned mirroring is disabled, as shown in Figure 5-12.

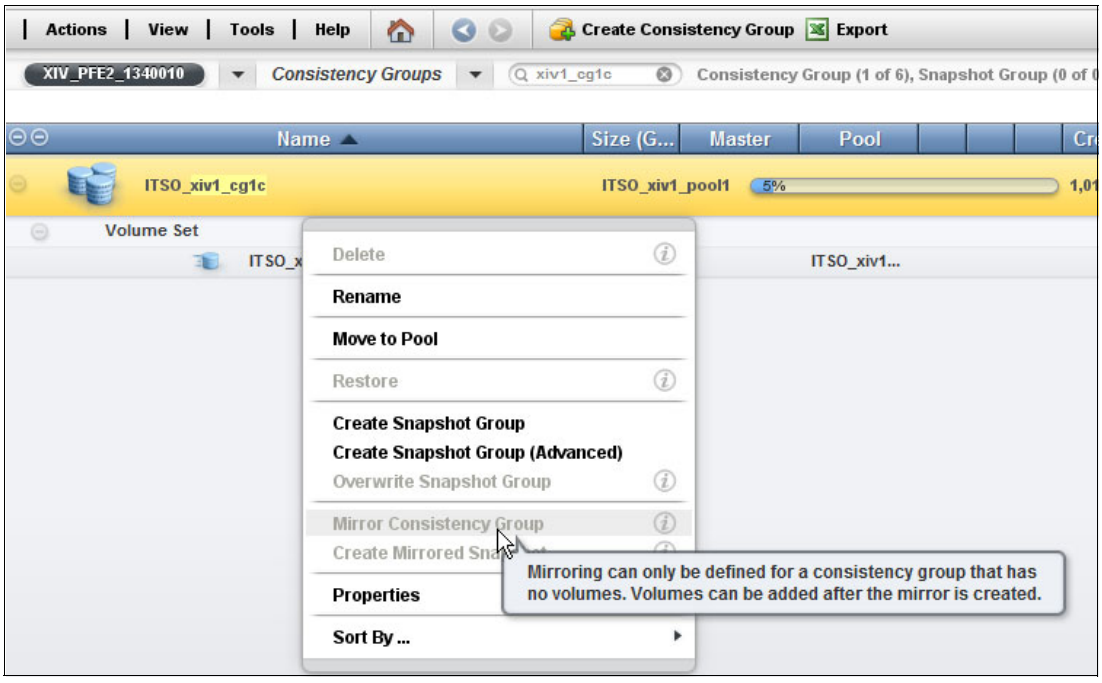


Figure 5-12 Message shown if a CG has volumes

3. Remove any volume from the CG then continue with Create Mirror dialog (Figure 5-13). The peer CG must already exist on the target system. Further explanations to these options are in 5.2.2, “Using the GUI for volume mirror activation” on page 118.

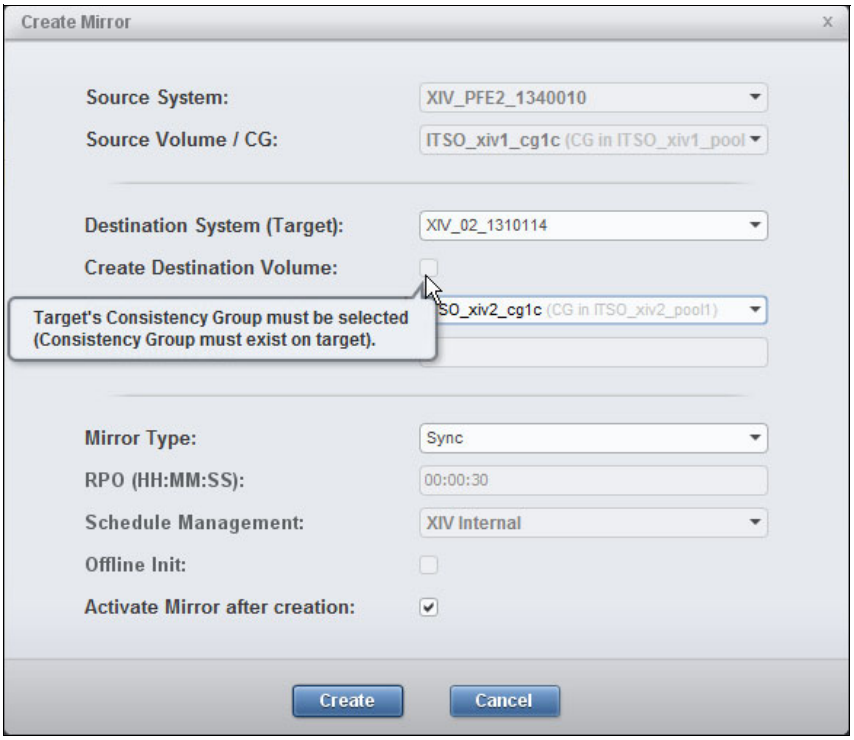


Figure 5-13 Setup CG for synchronous mirroring

4. Select **Pools** → **Volumes by Pools** → **Add to Consistency Group** (Figure 5-14) to add volumes to the consistency group.

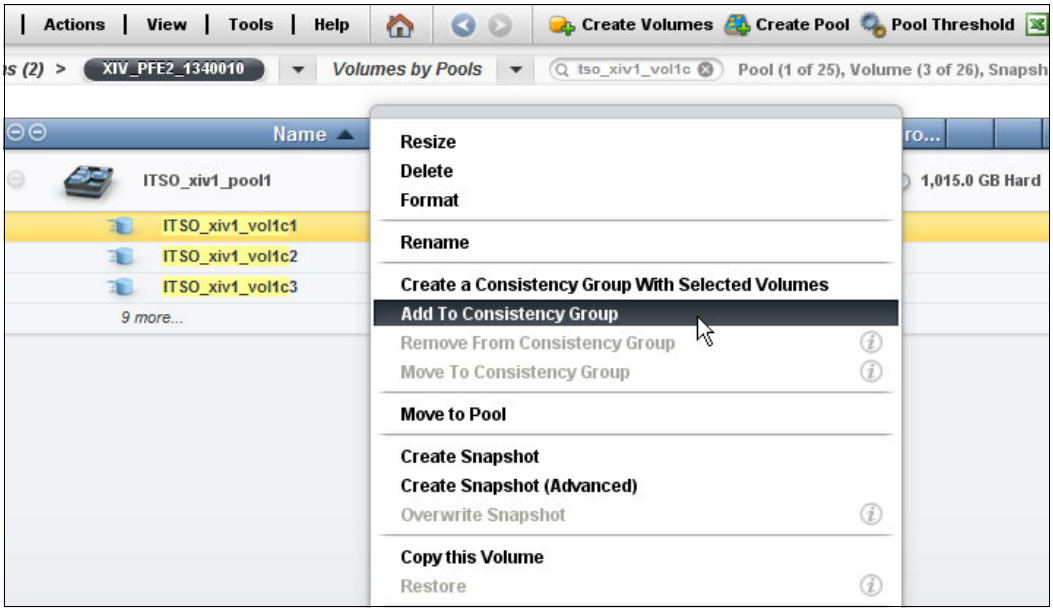


Figure 5-14 Adding volumes to a CG

Each volume must already be mirrored, otherwise an error like one shown in Figure 5-15 is issued when you try to add an unmirrored volume to a mirrored CG.

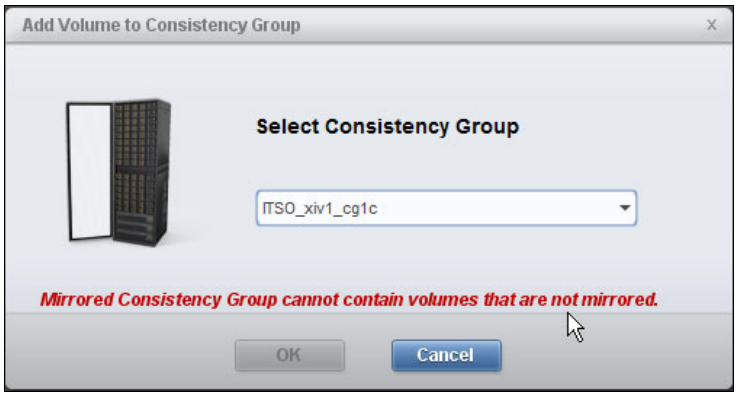


Figure 5-15 Error if adding unmirrored volumes to a mirrored CG

5. Mirror the individual volume, then add it to the mirrored CG, as depicted in Figure 5-16 and Figure 5-17 on page 125.

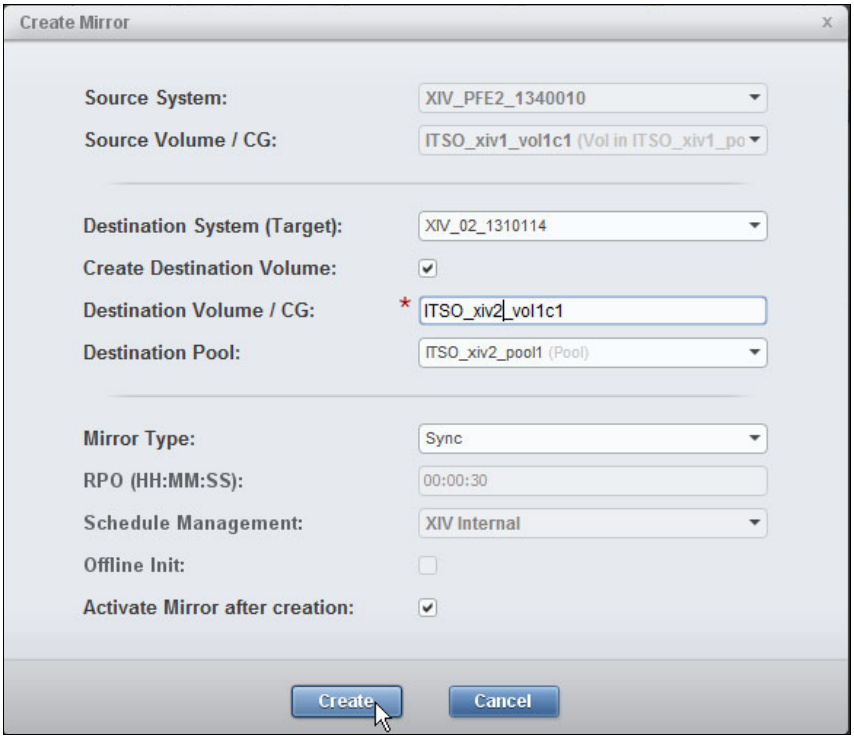


Figure 5-16 Each volume needs to be mirrored

Figure 5-17 Adding a mirrored volume to a mirrored CG

6. One more prerequisite must be met: the CG and the volume must have the same mirroring state, but *inactive*, otherwise an error occurs, as shown in Figure 5-18.

Failed
Operation failed!
Error: Volumes under Consistency Group Mirror should have the same mirroring activation state.
OK

Figure 5-18 Volumes and CG have different mirroring states

Mirror state of the CG is inactive (Figure 5-19).

Figure 5-19 CG mirroring state is inactive

7. After activating mirroring of the CG, the volume can be added (Figure 5-20).

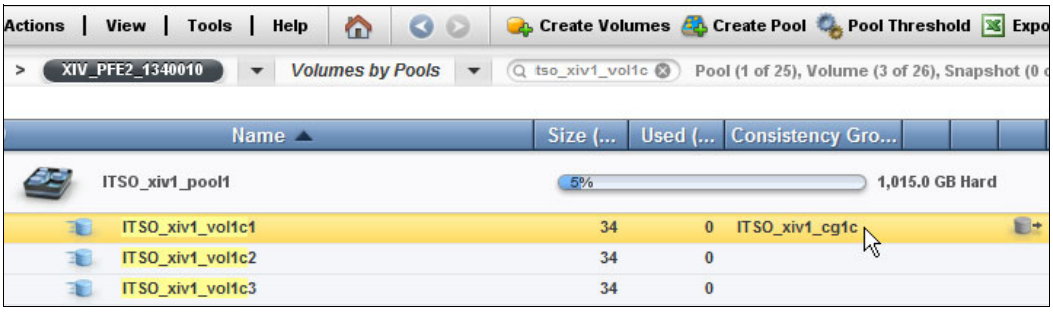


Figure 5-20 Volume successfully added to the mirrored CG

5.4 Mirrored snapshots (ad-hoc sync jobs)

An additional feature available to users of both synchronous and asynchronous mirroring is *mirrored snapshots*, also referred to as *ad-hoc sync jobs*. To explain further, the mirrored snapshots feature creates snapshots of the respective coupling peers at both local and remote sites, so that the local and remote snapshots are identical. Mirrored snapshots are manually initiated by the user and managed by the master peer in a mirroring relationship. This feature can be used to create application-consistent replicas, typically for backup and disaster recovery purposes, if the application has prepared for it.

After the snapshots are created, they can be managed independently. This means the local mirror snapshot can be deleted without affecting its remote peer.

5.4.1 Using the GUI for creating a mirrored snapshot

Use the following steps to create a mirrored snapshot:

1. Ensure the mirror is established and the status is synchronized as shown in Figure 5-21.

Figure 5-21 Mirror status is synchronized

2. At the production site, put the application into backup mode. This does not mean stopping the application, but it does mean having the application flush its buffers to disk so that a snapshot will contain application-consistent data. This step might momentarily reduce the I/O performance to the storage system.

3. Select one or more volumes or CGs, then use the context menu to select **Create Mirrored Snapshot** (Figure 5-22).

Figure 5-22 Create Mirrored Snapshot

4. A Snap and Sync dialog opens as shown in Figure 5-23, where the names of the mirrored snapshots can be entered. Click **Sync**.

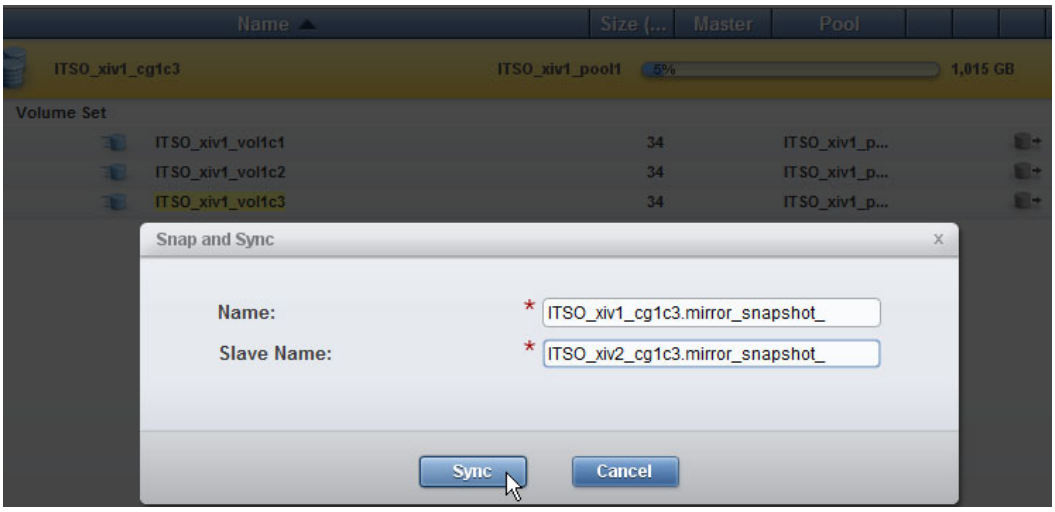


Figure 5-23 Snap and Sync pop-up window

5. Verify that mirrored snapshots are generated on both local and remote systems, as shown in Figure 5-24 and Figure 5-25.

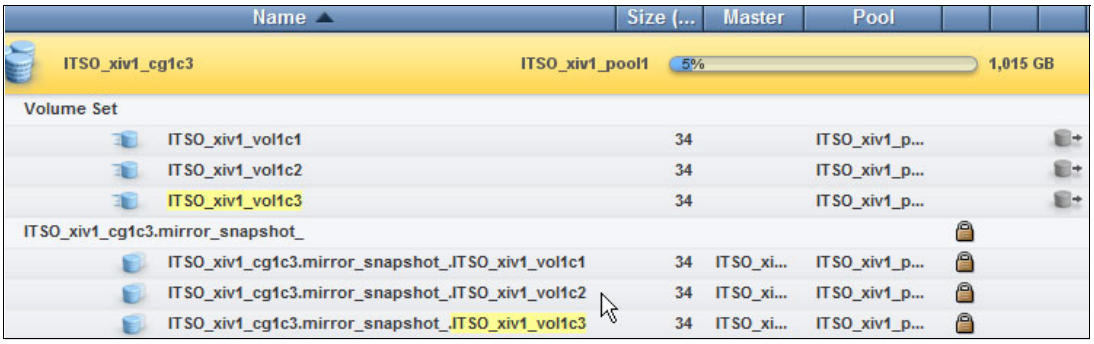


Figure 5-24 Mirror snapshots (master XIV)

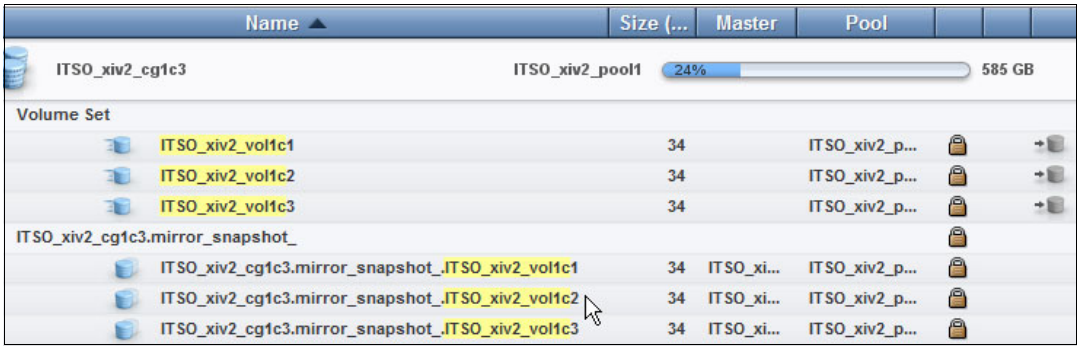


Figure 5-25 Mirror snapshots (slave XIV)

6. Next, you can take the production site application out of backup and restart it.
7. On the DR site, the mirrored snapshots can be unlocked and mapped to the local hosts for testing purposes.

5.5 Mirror activation, deactivation, and deletion

Mirroring can be manually activated and deactivated per volume or CG pair. When it is activated, the mirror is in *active* mode. When it is deactivated, the mirror is in *inactive* mode. These modes have the following functions:

- Active: Mirroring is functioning. Data written to the primary system is propagated to the secondary system.
- Inactive: Mirroring is deactivated. The data is not being written to the slave peer, but writes to the master volume are being recorded and can later be synchronized with the slave volume.

The mirror has the following characteristics:

- When a mirror is created, it is in inactive mode by default.
- A mirror can be deleted only when it is in inactive mode. Deleting removes all information associated with the mirror.
- A consistency group can be deleted only if it does not contain any volumes associated with it.

- If a mirror is deleted, the slave volume is *locked,* which means that it is write-protected. To enable writing select **Unlock** from the volume's context menu.
- Transitions between active and inactive states can be performed only from the IBM XIV that contains the master peer.
- In a situation where the primary IBM XIV becomes unavailable, execution of a role change transitions the slave peers at the secondary system to a master peer role so that work can resume at the secondary. When the primary is recovered, and before the mirror is resumed, the original master peer must be changed to a slave through a **`mirror_change_role`** command on the primary.

Note: The slave volume must be formatted before it can be part of a new mirror. Formatting also requires that all snapshots of the volume be deleted. However, formatting is not required if **Offline Init** is selected when you create a mirror.

5.6 Role reversal tasks (switch or change role)

With synchronous mirroring, roles can be modified by either *switching* or *changing* roles.

- *Switching* roles must be initiated on the master volume/CG when remote mirroring is operational. As the task name implies, it switches from master to slave role on one site and from slave to master role on the peer site.
- *Changing* roles can be performed at any time (when a pair is active or inactive) for the slave, and for the master when the coupling is inactive. A change role reverts only the role of that peer.

5.6.1 Switch roles

Switching roles exchanges the roles of master and slave volumes or CGs. It needs to be performed from the master peer and requires the pair to be synchronized as can be seen on Figure 5-26 on page 130. After switching roles, the master volume or CG becomes the slave volume or CG and vice versa. The operation can be executed from the GUI or by the **`mirror_switch_roles`** XCLI command.

There are two typical reasons for switching roles:

- Drills / DR tests

  Drills can be performed to test the functionality of the secondary site. In a drill, an administrator simulates a disaster and tests that all procedures are operating smoothly and that documentation is accurate.

- Scheduled maintenance

  To perform maintenance at the primary site, operations can be switched to the secondary site prior to the maintenance. This switch over cannot be performed if the master and slave volumes or CG are not synchronized.

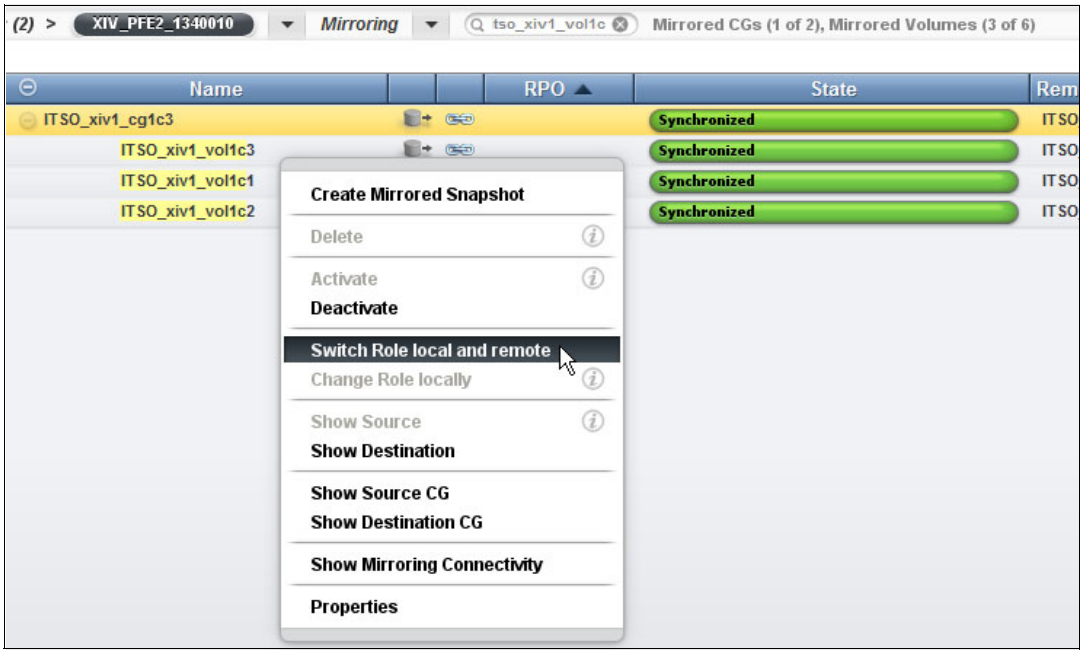


Figure 5-26 Switch Role initiated on master consistency group

Figure 5-27 demonstrates that *Switch Role* is not available on the slave peer site:

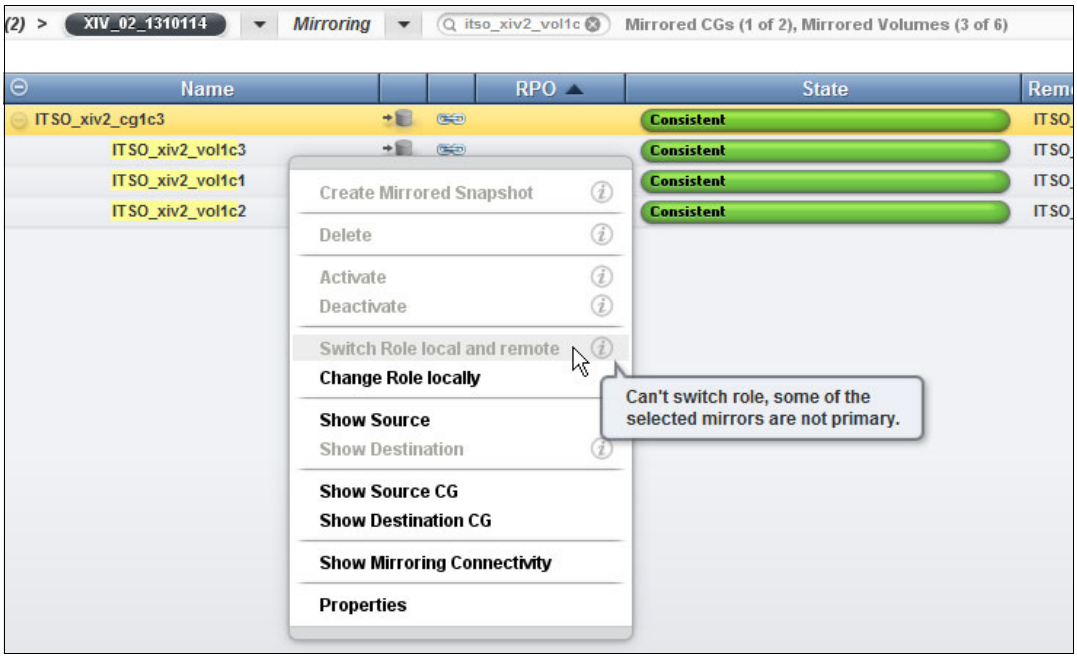


Figure 5-27 Switch role not active on slave peer

Normally, switching roles requires shutting down the applications at the primary site first, changing SAN zoning and XIV LUN mapping to allow access to the secondary site volumes, and then restarting the applications with access to the secondary IBM XIV. However, in certain clustered environments, this process is usually automated.

Note: Before the **`mirror_switch_roles`** command switches roles, the system stops accepting new writes to the local volume. With synchronous mirrors the system will perform all pending writes, and only after all pending writes have been committed, the roles are switched.

5.6.2 Change role

In the event of a disaster at the primary site, a role change at the secondary site is required as part of the recovery actions.

Assuming that the primary site is down and the secondary site will become the main production site, changing roles is performed at the secondary (now production) site first. Later, when the primary site is up again and communication is re-established, the role at the primary site is changed to slave to establish remote mirroring from the secondary site back to the normal production primary site. After data has been synchronized from the secondary site to the primary site, a switch role can be performed to again make the primary site the master.

Changing the slave peer role

The role of the slave volume/CG can be changed to the master role, as shown in Figure 5-28. After the change, the following situation is true:

- The slave volume/CG is now the master.
- The coupling has the status of *unsynchronized*.
- The coupling remains inactive, meaning that the remote mirroring is deactivated. This ensures an orderly activation when the role of the peer on the other site is changed.

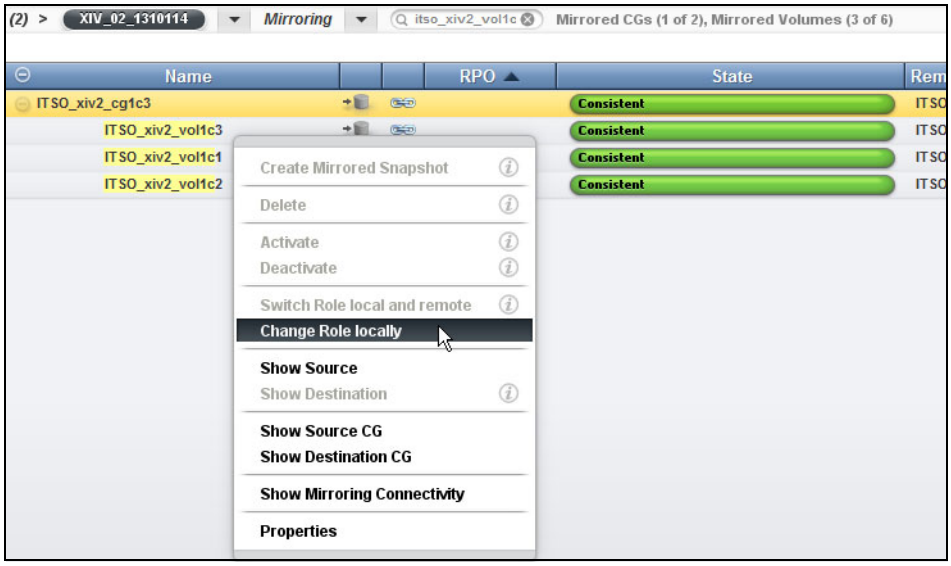


Figure 5-28 Change Role of a slave consistency group

The new master volume/CG at the secondary site starts to accept write commands from local hosts. Because coupling is not active, as is the case with any master volume in a mirror, metadata maintains a record of which write operations must be sent to the slave volume when communication resumes.

After changing the slave to the master, an administrator must change the original master to the slave role before communication resumes. If both peers are left with the same role of master, mirroring cannot be restarted.

Slave peer consistency

When a user is changing the slave volume/CG to a master volume or master consistency group and a last consistent snapshot exists, which was created because of a failed link during the process of resynchronizing, the system reverts the slave to the *last consistent snapshot.* See 5.7.1, “Last-consistent snapshot (LCS)” on page 133 for more information about the last consistent snapshot.

In addition, if a last-consistent snapshot exists and the role is changed from slave to master, the system automatically generates a snapshot of the volume. This snapshot is named most-updated snapshot. It is generated to preserve the latest version of the volume.

It is then up to the administrator to decide whether to use the most-updated, which might be inconsistent, or the last-consistent snapshot.

Changing the master peer role

When coupling is inactive, the master volume/CG can change its role. After such a change, the master volume/CG becomes the slave volume/CG.

All changes made to the master since the last time the peers were synchronized will be reverted to their original value. The master ceases serving host requests, and is set to accept replication from the other peer as a slave. If the command was issued when the link was unavailable, a most-updated snapshot of the master peer will be taken to capture the most recent changes that have not yet been replicated to the other peer.

Reconnection if both peers have the same role

Situations where both sides are configured to the same role can occur only when one side was changed. The roles must be changed to have one master and one slave. Change the volume roles as appropriate on both sides before the link is resumed.

If the link is resumed and both sides have the same role, the coupling will not become operational. To solve this problem, you must use the *change role* function on one of the volumes and then activate the coupling.

5.7 Link failure and last consistent snapshot

A synchronous mirror relationship has, by its nature, identical data on both local and remote with zero RPO. This principle is not maintainable if the link between the two sites is broken. In that case, data is written to only one site and needs to be resynchronized to the other site when connectivity is regained. During the time of resynchronization, data on both sites is not consistent. That is why you must take precautions to protect against a failure during the resynchronization phase. The means to preserve consistency is to generate a last-consistent snapshot (LCS) on the slave XIV after the link is regained and before resynchronization of any new data.

The following scenarios are samples:

- Resynchronization can be performed in any direction if one peer has the master role and the other peer has the slave role. If there was only a temporary failure of all links from the primary XIV to the secondary XIV, mirrors should be re-established with the original direction after links are operational again.

- If there was a disaster and production was moved to the secondary site, mirroring must be established firstly in direction from the secondary site to the primary site. This assures that changes to the secondary volumes during the outage are synchronized back to the primary site. Thereafter the direction could be changed again from the production to the DR site.
- A disaster recovery drill on the secondary site will often require resetting the changes applied there during the test and then resynchronizing the mirror from primary to secondary.

5.7.1 Last-consistent snapshot (LCS)

Before a resynchronization process is initiated, the system creates a snapshot of the slave volumes/CGs. A snapshot is created to ensure the usability of the slave volume/CG if the primary site experiences a disaster during the resynchronization process. If the master volume/CG is destroyed before resynchronization is completed, the slave volume/CG might be inconsistent, because it might have been only partially updated with the changes that were made to the master volume. To handle this situation, the secondary IBM XIV always creates a snapshot of the last consistent slave volumes/CGs after reconnecting to the primary XIV and before starting the resynchronization process. This special snapshot is called the *last-consistent snapshot* (LCS).

No LCS is created for couplings that are in an initialization state. The one or more snapshots are preserved until a volume/CG is completely synchronized. Then, it is deleted automatically, unless the slave peer role has changed during resynchronization.

If there is a disaster at the primary (master) site, the snapshot taken at the secondary (slave) site can be used to restore the slave volume/CG to a consistent state for production.

Important: The mirror relation at the secondary site must be deleted before the last consistent snapshot can be restored to the target volume/CG.

Tips:

- The last consistent snapshot can be deleted manually by using the **`vol_delete_mirror_snapshots`** XCLI command (by IBM support team only).
- Mirroring can be configured so that no last consistent snapshot is generated. This is useful when the system that contains the secondary volume is fully utilized and an additional snapshot cannot be created. The XCLI command to be used for this is **`pool_config_snapshots`** (by IBM support team only).

5.7.2 Last-consistent snapshot timestamp

A timestamp is taken when the coupling between the primary and secondary volumes becomes non-operational. This timestamp specifies the last time that the secondary volume was consistent with the primary volume.

This status has no meaning if the coupling's synchronization state is still Initialization. For synchronized couplings, this timestamp specifies the current time. Most important, for unsynchronized couplings, this timestamp denotes the time when the coupling became non-operational.

Figure 5-29 shows a last consistent snapshot during a mirror resync phase.

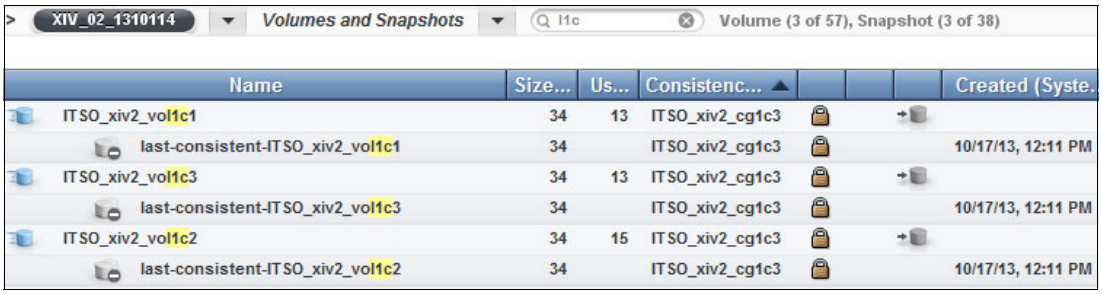


Figure 5-29 Last-consistent snapshot created during resync

Tip: The Created (System Time) column is not displayed by default. Right-click anywhere in the dark blue column heading area and move the Created (System Time) item from the Hidden Columns to the Visible Columns list before clicking update. This can be a helpful addition to the view output screen for configuration and management purposes.

5.7.3 External last consistent snapshot (ELCS)

Prior to the introduction of the external last consistent snapshot (ELCS), whenever a volume's role was changed back to slave and sometime whenever a new resynchronization process had started, the system would detect an LCS on the peer and would not create a new one. If, during such an event, the peer was not part of a mirrored consistency group (mirrored CG) it would mean that not all volumes have the same LCS timestamp. If the peer was part of a mirrored consistency group, we would have a consistent LCS but not as current as possibly expected. This situation is avoided with the introduction of the ELCS.

Whenever the role of a slave with an LCS is changed to master while mirroring resynchronization is in progress (in the system/target not specific to this volume), the LCS is renamed external last consistent (ELCS). The ELCS retains the LCS deletion priority of 0. If the peer's role is later changed back to slave and sometime afterwards a new resynchronization process starts, a new LCS will be created.

Subsequently changing the slave role again will rename the existing ELCS to external last consistent *x* (where *x* is the first available number starting from 1) and will rename the LCS to external last consistent. The deletion priority of external last consistent will be 0 (zero), but the deletion priority of the new external last consistent *x* will be the system default (1), and can thus be deleted automatically by the system upon pool space depletion.

It is crucial to validate whether the LCS or an ELCS (or even ELC x) should serve as a restore point for the slave peer volume if resynchronization cannot be completed. While snapshots with deletion priority 0 are not automatically deleted by the system to free space, the external last consistent and external last consistent x snapshots can be manually deleted by the administrator if required. Because the deletion of such snapshots might leave an inconsistent peer without a consistent snapshot from which to be restored (in case the resynchronization cannot complete as a result of master unavailability). it should generally be avoided even when pool space is depleted, unless the volume is guaranteed to be consistent.

5.8 Disaster recovery cases

There are two broad categories of disaster: one that destroys the primary site including data and one that causes unavailability of the primary site and its data. However, within these broad categories, several other situations might exist. Of the possible situations, the following list describes several scenarios in more detail:

- The XIV at the primary site is unavailable but the site itself and the servers are available.

  In this scenario, the volumes/CG on the XIV at the secondary site can be switched to master volumes/CG, servers at the primary site can be redirected to the XIV at the secondary site, and normal operations can start again. When the XIV at the primary site is recovered, the data can be mirrored from the secondary site back to the primary site. When the volume/CG synchronization is complete, the peer roles can be switched back to the master at the primary site, the slave at the secondary site, and the servers redirected back to the primary site.

- A disaster that causes the entire primary site and data to be unavailable.

  In this scenario, the standby inactive servers at the secondary site (if implemented) are activated and attached to the secondary XIV to continue normal operations. This requires changing the role of the slave volumes to become master volumes.

  After the primary site is recovered, the data at the secondary site can be mirrored back to the primary site to become synchronized again. A planned site switch can then take place to resume production activities at the primary site. See 5.6, “Role reversal tasks (switch or change role)” on page 129, for details related to this process.

- A disaster that breaks all links between the two sites but both sites remain running.

  In this scenario, the primary site continues to operate as normal. When the links are re-established the data from the primary site can be resynchronized to the secondary site. See 5.7, “Link failure and last consistent snapshot” on page 132, for more details.

5.8.1 Disaster recovery scenario with synchronous mirroring

In 5.1, “Synchronous mirroring considerations” on page 114, the steps required to set up, operate, and deactivate the mirror are addressed.

In this section, a scenario to demonstrate synchronous mirroring is covered. It describes the process under the assumption that all prerequisites are met to start configuring the remote mirroring couplings. In particular, the assumptions in this section are as follows:

- A host server exists and has volumes assigned at the primary site.
- Two IBM XIVs have been connected to each other over FC or iSCSI.
- A standby server exists at the secondary site.

Note: When you use XCLI commands, *quotation marks* must be used to enclose names that include spaces (as in `"volume 1"`). If they are used for names without spaces, the command still works. The examples in this scenario contain a mixture of commands with and without quotation marks.

The scenario describes the following phases:

1. Phase 1: Setup and configuration

   Perform initial setup, activate coupling, write data to three volumes, and prove that the data has been written and that the volumes are synchronized.

2. Phase 2: Simulating a disaster on the primary site

   The link is broken between the two sites to simulate that the primary site is unavailable. First, the slave volumes, which are mapped to the standby server at the secondary site, are changed to master volumes and new data is written on them.

3. Phase 3: Recovering the primary site

   The master volumes at the primary site are changed to slave volumes and data is mirrored back from the secondary site to the primary site.

4. Phase 4: Switching production back to the primary site (failback to the primary site)

   When the data is synchronized, the volume roles are switched back to the original roles (that is, master volumes at the primary site and slave volumes at the secondary site) and the original production server at the primary site is used.

Phase 1: Setup and configuration

In the sample scenario, on the primary site, a server with three volumes in a CG is being used and two IBM XIVs are defined as mirroring targets.

After the couplings have been created and activated (explained in 5.1, “Synchronous mirroring considerations” on page 114), the environment resembles that illustrated in Figure 5-30.

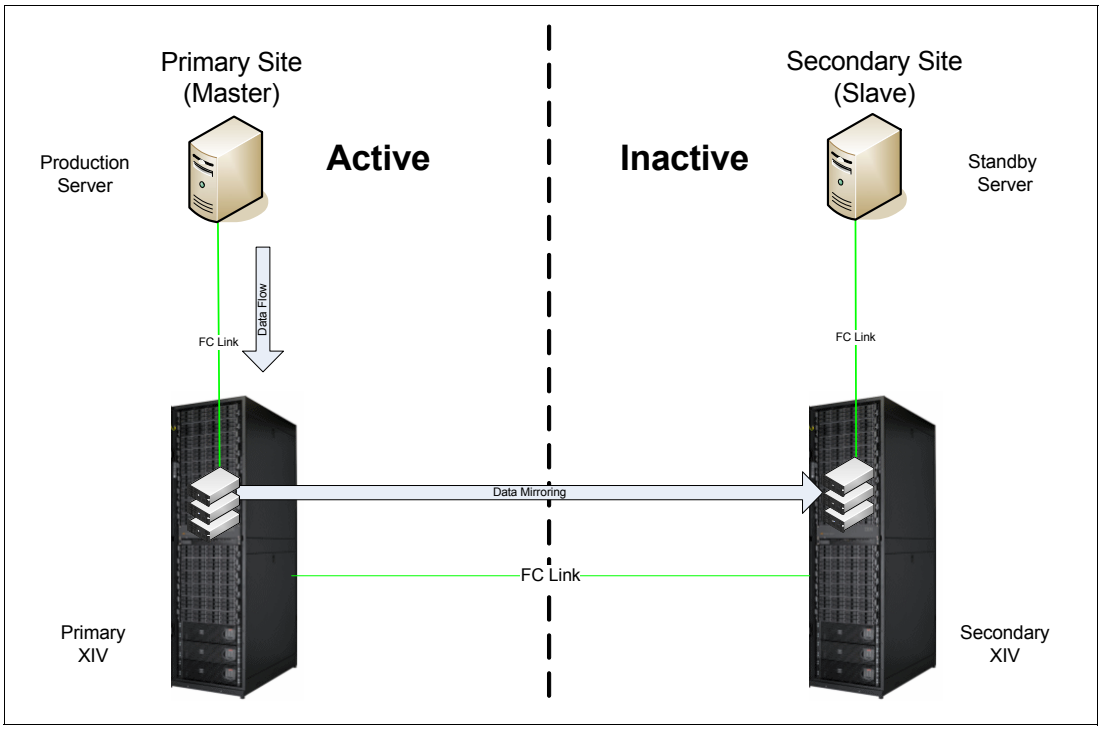


Figure 5-30 Environment with remote mirroring activated

The current data on the production server is shown in Example 5-7 on page 137.

Example 5-7 Production server data

```
bladecenter-h-prod:~ # ll /mpathi
total 8010808
-rw-r--r-- 1 root root 1024000000 Oct 16 18:22 file_i_1GB
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_i_1GB_2
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_i_1GB_3
-rw-r--r-- 1 root root 2048000000 Oct 16 18:56 file_i_2GB_1
-rw-r--r-- 1 root root 2048000000 Oct 16 19:03 file_i_2GB_2

bladecenter-h-prod:~ # ll /mpathj
total 8010808
-rw-r--r-- 1 root root 1024000000 Oct 16 18:22 file_j_1GB
-rw-r--r-- 1 root root 1024000000 Oct 16 18:43 file_j_1GB_2
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_j_1GB_3
-rw-r--r-- 1 root root 2048000000 Oct 16 18:56 file_j_2GB_1
-rw-r--r-- 1 root root 2048000000 Oct 16 19:03 file_j_2GB_2
```

Phase 2: Disaster simulation at primary site

This phase of the scenario simulates a disaster at the primary site. All communication has been lost between the primary and secondary sites because of a complete power failure or a disaster. This is depicted in Figure 5-31.

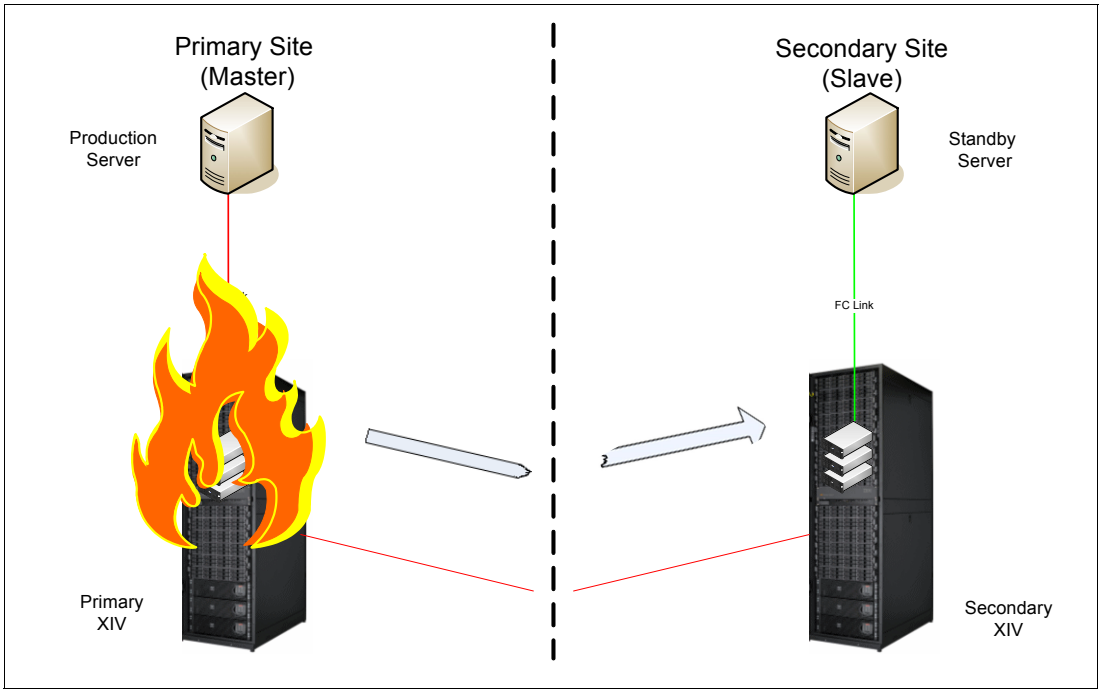


Figure 5-31 Primary site disaster

Change role at the secondary site using the GUI or XCLI

On the secondary IBM XIV, go to the Remote Mirroring menu, right-click on the CG, and select **Change Role locally** (Figure 5-32).

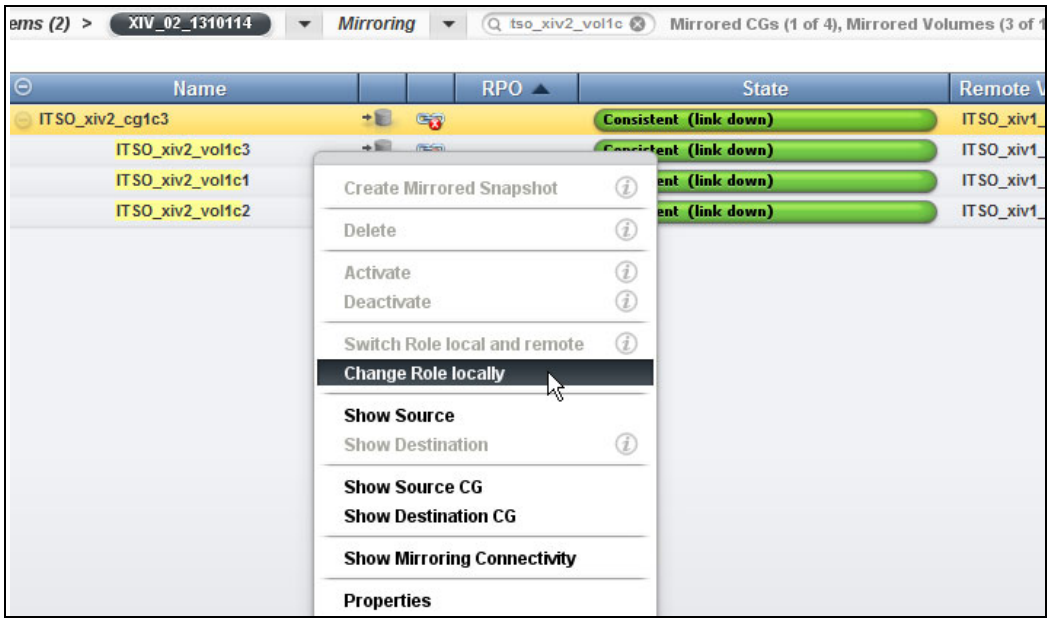


Figure 5-32 Remote mirror change role

The figure shows that the synchronization status is still *Consistent (link down)* for the couplings that are yet to be changed. The reason is because this is the last known state. When the role is changed, the coupling is automatically deactivated and reported as inactive in the GUI.

The same change can be achieved using the XCLI. Use the following steps to change roles for the slave volumes at the secondary site and make them master volumes so that the standby server can write to them:

1. On the secondary IBM XIV, open an XCLI session and run the **`mirror_change_role`** command (Example 5-8).

Example 5-8 Remote mirror change role

```
XIV_02_1310114>>mirror_list cg=ITSO_xiv2_cg1c3
Name            Mirror Type      Mirror Object Role   Remote System      Remote Peer    Active   Status        Link Up
ITSO_xiv2_cg1c3 sync_best_effort CG              Slave  XIV_PFE2_1340010  ITSO_xiv1_cg1c3   yes  Consistent    no

XIV_02_1310114>>mirror_change_role cg=ITSO_xiv2_cg1c3
Warning:   ARE_YOU_SURE_YOU_WANT_TO_CHANGE_THE_PEER_ROLE_TO_MASTER y/n: y
Command executed successfully.
```

2. To view the status of the coupling run the **`mirror_list`** command shown in Example 5-9. This example shows that after the role is changed, the coupling is automatically deactivated and in the XCLI the status is reported as *unsynchronized*.

Example 5-9 List mirror couplings

```
XIV_02_1310114>>mirror_list cg=ITSO_xiv2_cg1c3
Name             Mirror Type ´    Mirror Object Role   Remote System    Remote Peer      Active  Status         Link Up
ITSO_xiv2_cg1c3  sync_best_effort CG            Master XIV_PFE2_1340010  ITSO_xiv1_cg1c3  no    Unsynchronized no
XIV_02_1310114>>
```

Map volumes to standby server

At this point, map the relevant mirrored volumes to the standby server (Figure 5-33). After the volumes are mapped, continue working as normal.

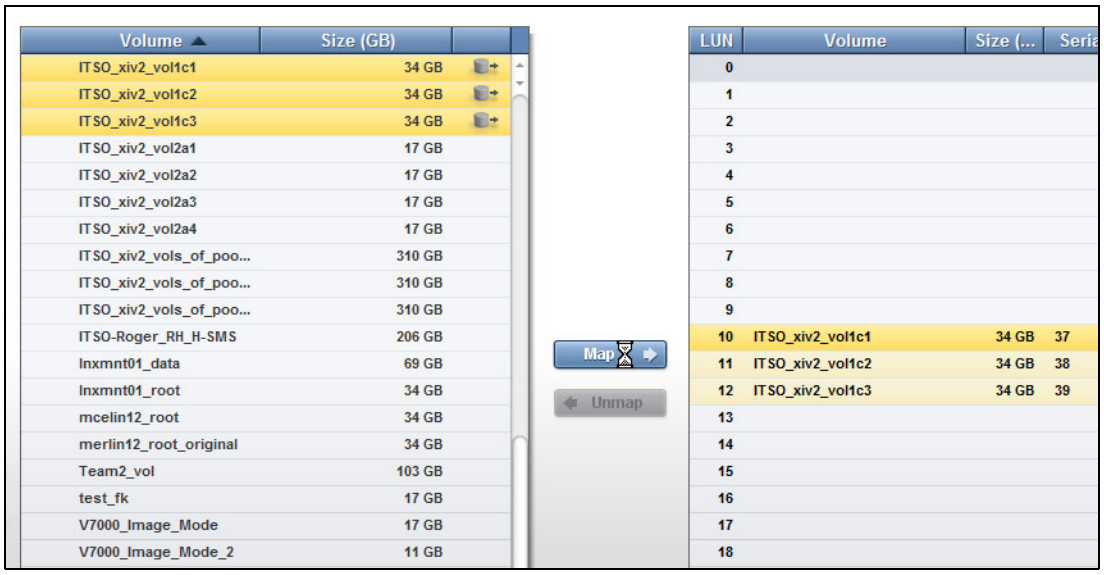


Figure 5-33 Additional data added to the standby server

Environment with production now at the secondary site

Figure 5-34 illustrates production at the secondary site.

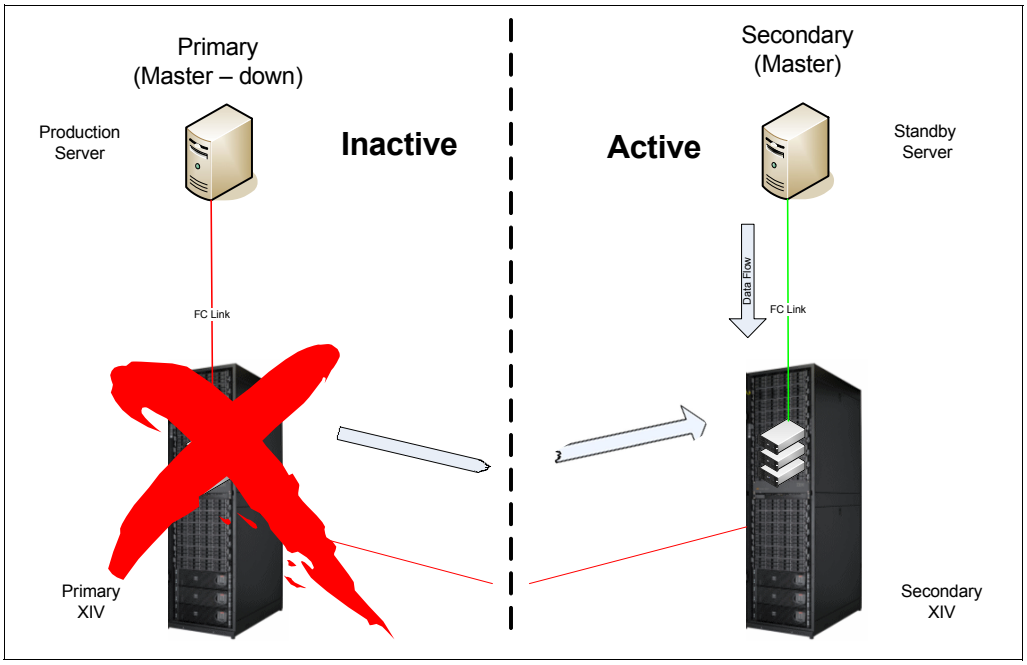


Figure 5-34 Production at secondary site

Continue working with standby server

Continue to work on standby server, demonstrated by adding new files (Example 5-10).

Example 5-10 Data added to standby server after switchover

```
bladecenter-h-standby:~ # ll /mpathp
total 11010808
-rw-r--r-- 1 root root 1024000000 Oct 16 18:22 file_j_1GB
-rw-r--r-- 1 root root 1024000000 Oct 16 18:43 file_j_1GB_2
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_j_1GB_3
-rw-r--r-- 1 root root 2048000000 Oct 16 18:56 file_j_2GB_1
-rw-r--r-- 1 root root 2048000000 Oct 16 19:03 file_j_2GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:39 file_p_1GB_1
-rw-r--r-- 1 root root 1048576000 Oct 21 16:44 file_p_1GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:44 file_p_1GB_3

bladecenter-h-standby:~ # ll /mpathq
total 11010808
-rw-r--r-- 1 root root 1024000000 Oct 16 18:22 file_i_1GB
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_i_1GB_2
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_i_1GB_3
-rw-r--r-- 1 root root 2048000000 Oct 16 18:56 file_i_2GB_1
-rw-r--r-- 1 root root 2048000000 Oct 16 19:03 file_i_2GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:49 file_q_1GB_1
-rw-r--r-- 1 root root 1048576000 Oct 21 16:49 file_q_1GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:49 file_q_1GB_3
```

Phase 3: Recovery of the primary site

During this phase, the primary site will be recovered after communication between the primary and secondary site has been regained. The assumption is that there was no damage to the primary site and the data from before the breakdown is still available for resynchronization. New data from the standby server had been written to the secondary IBM XIV. At the primary site the original production server is still off, as illustrated in Figure 5-35.

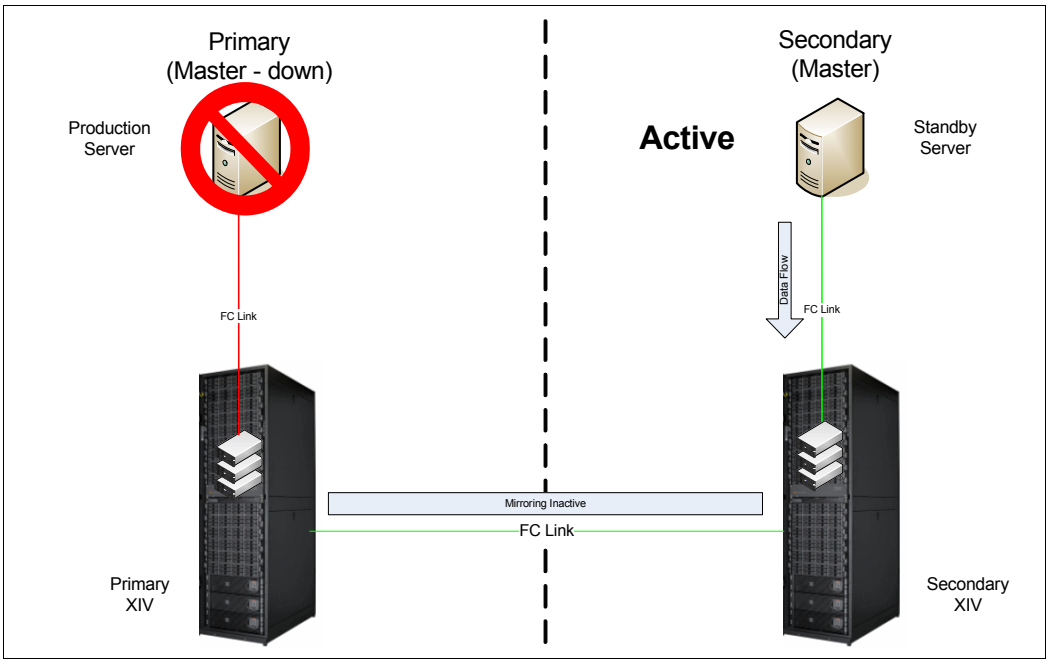


Figure 5-35 Primary site recovery

Change role at the primary site using the GUI

Change volumes/CGs at the primary site from master to slave roles.

Attention: Prior to executing the change role, verify that the original production server is not accessing the volumes. Either stop the server or unmap its volumes.

Complete the following steps:

1. On the primary XIV, go to the Remote Mirroring menu. The synchronization status is probably *Inactive*. Select one or more couplings or a CG, right-click, and select **Change Role locally**, as shown in Figure 5-36.

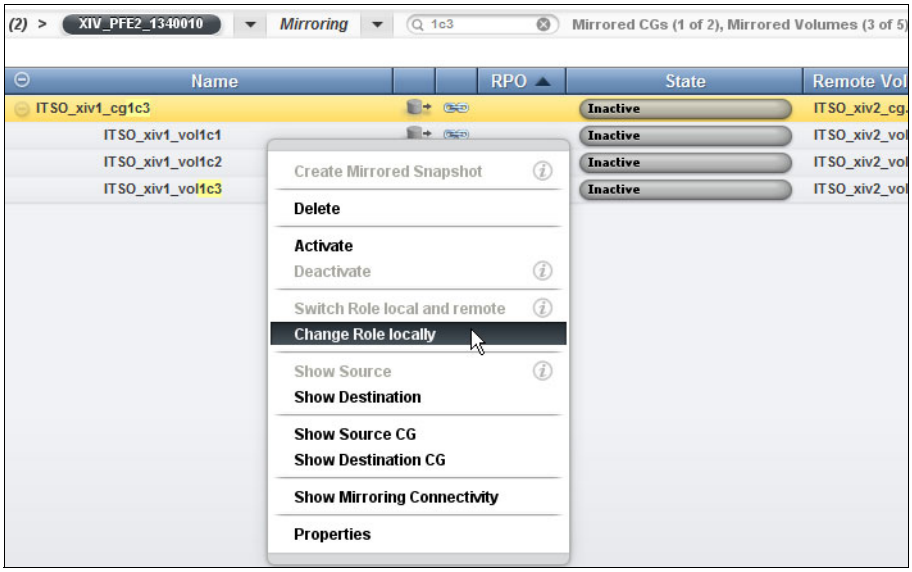


Figure 5-36 Change master volumes to slave volumes on the primary XIV

2. When prompted, click **OK** to confirm the role change.

   The role is changed to destination (slave), as Figure 5-37 shows.

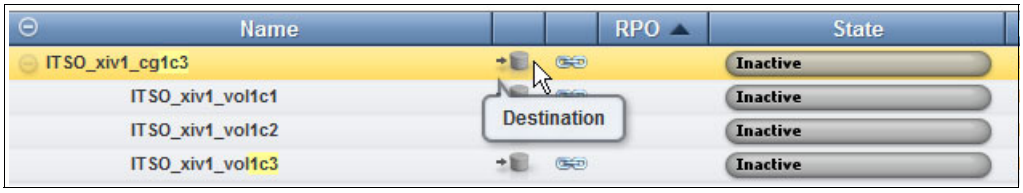


Figure 5-37 New role as slave volume

3. Repeat steps 1 - 2 for all remaining volumes/CGs that must be changed.

Change role at the primary site using XCLI

Change volumes/CGs at the primary site from master to slave roles using XCLI command.

Attention: Before doing the followings steps, ensure that the original production server is not accessing the volumes. Either stop the server or unmap its volumes.

1. On the primary XIV, open an XCLI session and run the **`mirror_change_role`** command as shown in Example 5-11 on page 142.

Example 5-11 Change master volumes to slave volumes on the primary XIV

```
XIV_PFE2_1340010>>mirror_change_role cg=ITSO_xiv1_cg1c3 new_role=slave
Warning:   ARE_YOU_SURE_YOU_WANT_TO_CHANGE_THE_PEER_ROLE_TO_SLAVE y/n: y
Command executed successfully.
```

2. To view the status of the coupling, run the **mirror_list** command; the result is shown in Example 5-12. Note that the XCLI status is *Inconsistent* but the GUI shows *Inactive.*

Example 5-12 List mirror couplings

```
XIV_PFE2_1340010>>mirror_list
Name              Mirror Type       Mirror Object Role   Remote System   Remote Peer       Active Status Link    Up
ITSO_xiv1_cg1c3   sync_best_effort  CG            Slave  XIV_02_1310114  ITSO_xiv2_cg1c3   no     Inconsistent   yes
ITSO_xiv1_vol1c1  sync_best_effort  Volume        Slave  XIV_02_1310114  ITSO_xiv2_vol1c1  no     Inconsistent   yes
ITSO_xiv1_vol1c2  sync_best_effort  Volume        Slave  XIV_02_1310114  ITSO_xiv2_vol1c2  no     Inconsistent   yes
ITSO_xiv1_vol1c3  sync_best_effort  Volume        Slave  XIV_02_1310114  ITSO_xiv2_vol1c3  no     Inconsistent   yes
```

3. Repeat steps 1 and 2 to change other couplings.

Reactivating mirroring on secondary site using the GUI

To reactivate the remote mirror coupling using the GUI, complete the following steps:

1. On the secondary XIV, select **Remote** → **Mirroring** and highlight all the couplings that you want to activate. Right-click and select **Activate** (Figure 5-38).

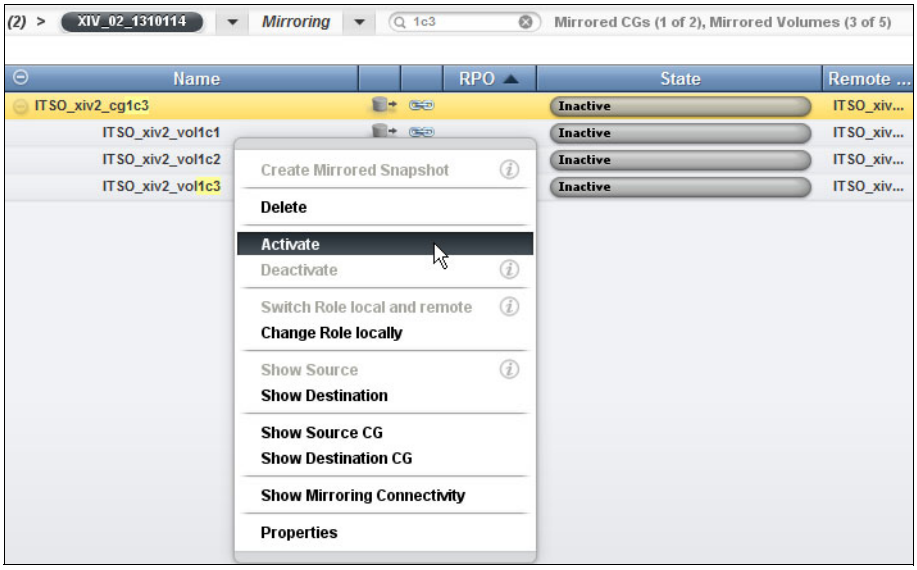


Figure 5-38 Reactivating mirroring on secondary

After activation mirroring, the status changes from *Inactive* to *Unsynchronized* (Figure 5-39).

Figure 5-39 Secondary / Master starts synchronisation

After synchronization is complete, the state of the primary slave is *Consistent* (Figure 5-40).

Figure 5-40 Primary slave is consistent

2. Repeat activating all required couplings until all volumes/CGs are done.

Reactivating mirroring on secondary site using XCLI

To reactivate the remote mirror coupling using the XCLI, complete the following steps:

1. Check mirroring status on secondary before activation using **`mirror_list`** (Example 5-13).

Example 5-13 Mirror status on secondary master before activation is Unsynchronized

```
XIV_02_1310114>> mirror_list
Name             Mirror Type       Mirror Object  Role   Remote System    Remote Peer     Active Status          Link Up
ITSO_xiv2_cg1c3  sync_best_effort  CG             Master XIV_PFE2_1340010 ITSO_xiv1_cg1c3  no   Unsynchronized   yes
ITSO_xiv2_vol1c1  sync_best_effort  Volume         Master XIV_PFE2_1340010 ITSO_xiv1_vol1c1 no   Unsynchronized   yes
ITSO_xiv2_vol1c2  sync_best_effort Volume         Master XIV_PFE2_1340010 ITSO_xiv1_vol1c2 no   Unsynchronized   yes
ITSO_xiv2_vol1c3  sync_best_effort Volume         Master XIV_PFE2_1340010 ITSO_xiv1_vol1c3 no   Unsynchronized   yes
```

2. Check mirroring status on primary before activation by using **`mirror_list`** (Example 5-14).

Example 5-14 Mirror status on primary slave before activation is Inconsistent

```
XIV_PFE2_1340010>> mirror_list
Name             Mirror Type       Mirror Object  Role   Remote System  Remote Peer      Active  Status        Link Up
ITSO_xiv1_cg1c3  sync_best_effort  CG             Slave  XIV_02_1310114 ITSO_xiv2_cg1c3   no     Inconsistent   yes
ITSO_xiv1_vol1c1 sync_best_effort  Volume         Slave  XIV_02_1310114 ITSO_xiv2_vol1c1  no     Inconsistent   yes
ITSO_xiv1_vol1c2 sync_best_effort  Volume         Slave  XIV_02_1310114 ITSO_xiv2_vol1c2  no     Inconsistent   yes
ITSO_xiv1_vol1c3 sync_best_effort  Volume         Slave  XIV_02_1310114 ITSO_xiv2_vol1c3  no     Inconsistent   yes
```

3. On the secondary IBM XIV, use **`mirror_activate`** as in Example 5-15.

Example 5-15 Reactivating the mirror coupling

```
XIV_02_1310114>>mirror_activate cg=ITSO_xiv2_cg1c3
Command executed successfully.
```

4. On the secondary XIV, run the **`mirror_list`** command to see the status of the couplings, as illustrated in Example 5-16.

Example 5-16 Mirror status on secondary master after activation

```
XIV_02_1310114>> mirror_list
Name             Mirror Type       Mirror Object  Role   Remote System     Remote Peer      Active  Status       Link Up
ITSO_xiv2_cg1c3  sync_best_effort  CG             Master XIV_PFE2_1340010  ITSO_xiv1_cg1c3  yes    Synchronized   yes
ITSO_xiv2_vol1c1 sync_best_effort  Volume         Master  XIV_PFE2_1340010  ITSO_xiv1_vol1c1 yes    Synchronized   yes
ITSO_xiv2_vol1c2 sync_best_effort  Volume         Master  XIV_PFE2_1340010  ITSO_xiv1_vol1c2 yes    Synchronized   yes
ITSO_xiv2_vol1c3 sync_best_effort  Volume         Master  XIV_PFE2_1340010  ITSO_xiv1_vol1c3 yes    Synchronized   yes
```

5. On the primary XIV run the **mirror_list** command to see the status of the couplings, as shown in Example 5-17.

Example 5-17 Mirror status on primary slave after activation

```
XIV_PFE2_1340010>> mirror_list
Name             Mirror Type      Mirror Object Role    Remote System   Remote Peer       Active   Status Link Up
ITSO_xiv1_cg1c3  sync_best_effort CG            Slave   XIV_02_1310114  ITSO_xiv2_cg1c3   yes      Consistent   yes
ITSO_xiv1_vol1c1 sync_best_effort Volume         Slave   XIV_02_1310114  ITSO_xiv2_vol1c1   yes     Consistent   yes
ITSO_xiv1_vol1c2 sync_best_effort  Volume        Slave   XIV_02_1310114  ITSO_xiv2_vol1c2   yes      Consistent   yes
ITSO_xiv1_vol1c3 sync_best_effort Volume        Slave   XIV_02_1310114  ITSO_xiv2_vol1c3   yes     Consistent   yes
```

6. Repeat steps 1 - 2 to activate additional couplings.

Environment with remote mirroring reactivated

Figure 5-41 illustrates production active at the secondary site, now the master. The standby server is running production with synchronous mirroring to the primary site, now the slave.

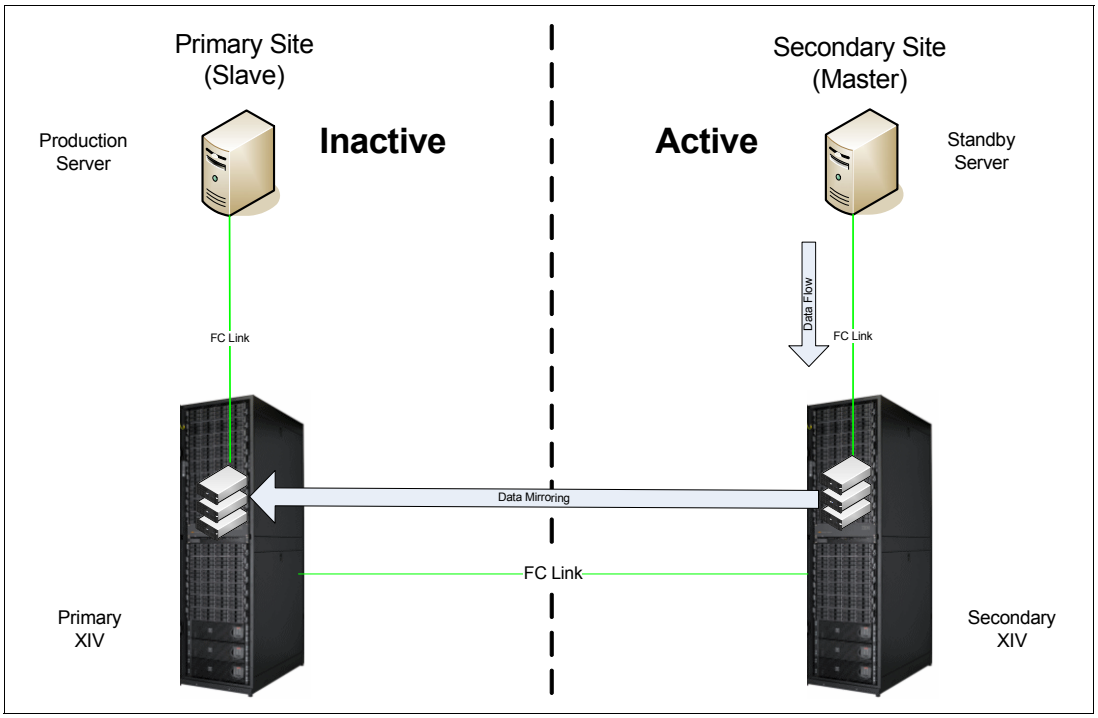


Figure 5-41 Mirroring reactivated

Phase 4: Switching production back to the primary site

At this stage, mirroring is reactivated with production at the secondary site but will be moved back to the original production site. The following steps are necessary to achieve this:

1. Applications on standby server are stopped and volumes are unmounted.
2. Switch roles of all volumes/CGs.
3. Switch from the standby server to the original production server.

Switching roles using the GUI

To switch the role using the GUI, complete the following steps:

1. At the secondary site, ensure that all the volumes of the standby server are synchronized. Stop the applications and unmount the volumes/CGs from the server.
2. On the secondary XIV, go to **Remote** → **Mirroring** menu, highlight the required coupling, and select **Switch Roles** (Figure 5-42).

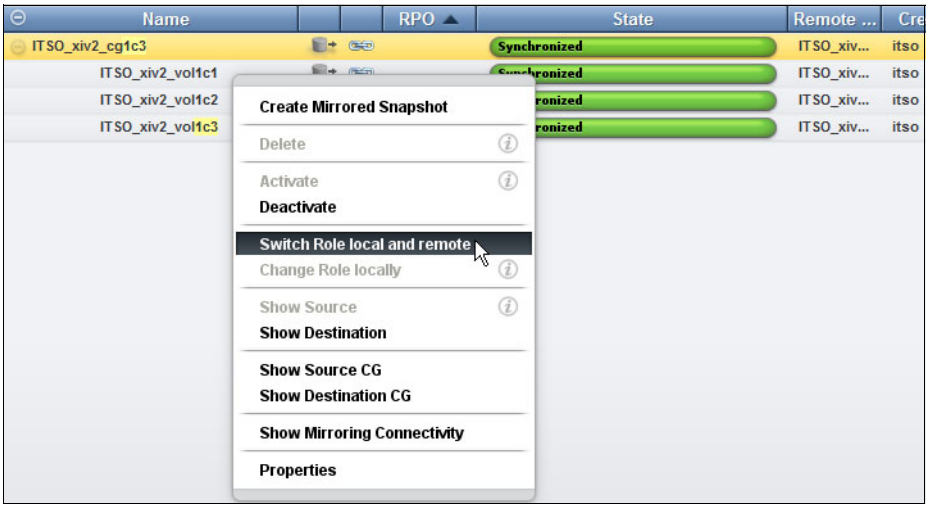


Figure 5-42 Switch roles on secondary master

3. You are prompted for confirmation (Figure 5-43). Click **OK**.

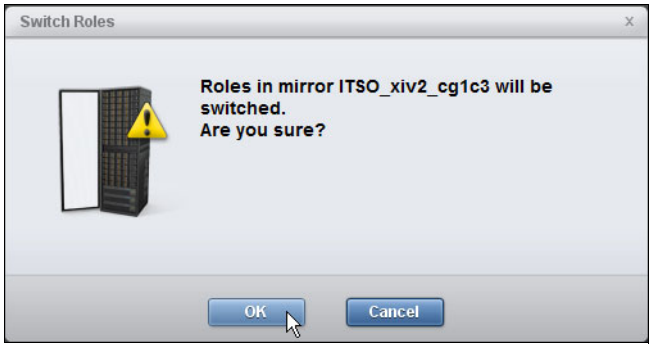


Figure 5-43 Confirm switching roles

The role is changed to slave (destination) and the mirror status is *Consistent* (Figure 5-44).

Figure 5-44 Role is switched to slave (destination) on secondary

4. Click **Remote** → **Mirroring** on the primary XIV and check the status of the coupling. The peer volume/CG role is now master (source) as shown in Figure 5-45.

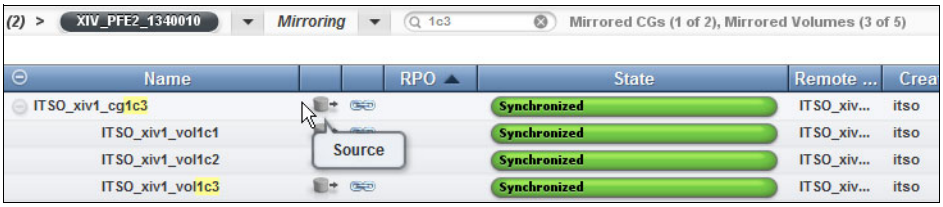


Figure 5-45 Switch role to master volume on the primary XIV

5. Reassign volumes back to the production server at the primary site and power it on again. Normal operation can resume.

Switching roles using XCLI

To switch over the role using XCLI, complete the following steps:

1. At the secondary site, ensure that all the volumes for the standby server are synchronized, and shut down the servers.
2. On the secondary XIV, open an XCLI session and run the **`mirror_switch_roles`** command (Example 5-18).

Example 5-18 Switch from master CG to slave CG on secondary IBM XIV

```
XIV_02_1310114>>mirror_switch_roles cg=ITSO_xiv2_cg1c3
Warning:   ARE_YOU_SURE_YOU_WANT_TO_SWITCH_ROLES y/n: y
Command executed successfully.
```

3. On the secondary XIV, run the **`mirror_list`** command to list the mirror coupling (Example 5-19).

Example 5-19 Mirror status on the secondary IBM XIV

```
XIV_02_1310114>> mirror_list
Name             Mirror Type      Mirror Object Role  Remote System    Remote Peer      Active Status Link  Up
ITSO_xiv2_cg1c3  sync_best_effort CG            Slave XIV_PFE2_1340010 ITSO_xiv1_cg1c3  yes    Consistent   yes
ITSO_xiv2_vol1c1 sync_best_effort Volume        Slave XIV_PFE2_1340010 ITSO_xiv1_vol1c1 yes    Consistent   yes
ITSO_xiv2_vol1c2 sync_best_effort Volume        Slave XIV_PFE2_1340010 ITSO_xiv1_vol1c2 yes    Consistent   yes
ITSO_xiv2_vol1c3 sync_best_effort Volume        Slave XIV_PFE2_1340010 ITSO_xiv1_vol1c3 yes    Consistent   yes
```

4. On the primary XIV, run the **`mirror_list`** command to list the mirror couplings (Example 5-20).

Example 5-20 Mirror statuses on the primary IBM XIV

```
XIV_PFE2_1340010>> mirror_list
Name             Mirror Type      Mirror Object Role   Remote System  Remote Peer      Active Status       Link Up
ITSO_xiv1_cg1c3  sync_best_effort CG            Master XIV_02_1310114 ITSO_xiv2_cg1c3  yes    Synchronized yes
ITSO_xiv1_vol1c1 sync_best_effort Volume        Master XIV_02_1310114 ITSO_xiv2_vol1c1 yes    Synchronized yes
ITSO_xiv1_vol1c2 sync_best_effort Volume        Master XIV_02_1310114 ITSO_xiv2_vol1c2 yes    Synchronized yes
ITSO_xiv1_vol1c3 sync_best_effort Volume        Master XIV_02_1310114 ITSO_xiv2_vol1c3 yes    Synchronized yes
```

5. Remount volumes back to the production server at the primary site and start it again. Continue in normal production mode. Example 5-21 shows that all new data is now available at the primary site.

Example 5-21 Data on production server after switch back

```
bladecenter-h-standby:~ # ll /mpathi
total 11010808
-rw-r--r-- 1 root root 1024000000 Oct 16 18:22 file_i_1GB
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_i_1GB_2
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_i_1GB_3
-rw-r--r-- 1 root root 2048000000 Oct 16 18:56 file_i_2GB_1
-rw-r--r-- 1 root root 2048000000 Oct 16 19:03 file_i_2GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:49 file_q_1GB_1
-rw-r--r-- 1 root root 1048576000 Oct 21 16:49 file_q_1GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:49 file_q_1GB_3

bladecenter-h-standby:~ # ll /mpathj
total 11010808
-rw-r--r-- 1 root root 1024000000 Oct 16 18:22 file_j_1GB
-rw-r--r-- 1 root root 1024000000 Oct 16 18:43 file_j_1GB_2
-rw-r--r-- 1 root root 1024000000 Oct 16 18:44 file_j_1GB_3
-rw-r--r-- 1 root root 2048000000 Oct 16 18:56 file_j_2GB_1
-rw-r--r-- 1 root root 2048000000 Oct 16 19:03 file_j_2GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:39 file_p_1GB_1
-rw-r--r-- 1 root root 1048576000 Oct 21 16:44 file_p_1GB_2
-rw-r--r-- 1 root root 1048576000 Oct 21 16:44 file_p_1GB_3
```

Environment is back to production state

The environment is now back to normal production state with mirroring from the primary site to the secondary site, as shown in Figure 5-46.

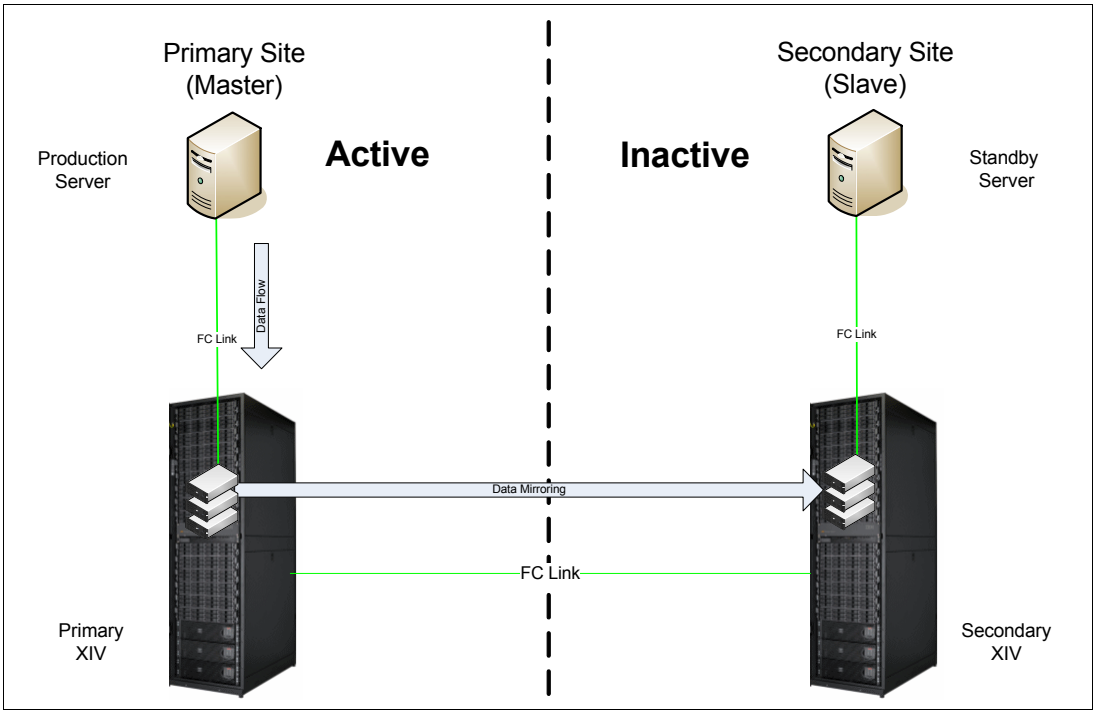


Figure 5-46 Environment back to production state

6

Asynchronous remote mirroring

This chapter describes the basic characteristics, options, and available interfaces for asynchronous remote mirroring. It also includes step-by-step procedures for setting up, running, and removing the mirror.

Asynchronous mirroring of a volume or a consistency group synchronization is attained through a periodic, recurring activity that takes a snapshot of a designated source and updates a designated target with differences between that snapshot and the last-replicated version of the source. XIV Storage System asynchronous mirroring supports multiple consistency groups with different recovery point objectives. XIV Storage System asynchronous mirroring supports up to 8 XIV targets, 512 mirrored pairs, scheduling, event reporting, and statistics collection.

Asynchronous mirroring enables replication between two XIV Storage System volumes or consistency groups (CG). It does not suffer from the latency inherent to synchronous mirroring, thereby yielding better system responsiveness and offering greater flexibility for implementing disaster recovery solutions.

Important: For mirroring, a reliable, dedicated network is preferred. Links can be shared, but require available and consistent network bandwidth. The specified minimum bandwidth (10 Mbps for FC and 50 Mbps for iSCSI for XIV software v11.1.x) is a functional minimum and does not necessarily guarantee what an acceptable replication speed will be achieved in a given customer environment and workload.

Also, minimum bandwidths are *not* time-averaged, as typically reported by network monitoring packages, but are instantaneous, constant requirements, typically achievable only through network quality of service (QoS) or similar.

© Copyright IBM Corp. 2014. All rights reserved.

6.1 Asynchronous mirroring configuration

The mirroring configuration process involves configuring volumes and CGs. When a pair of volumes or consistency groups point to each other, it is referred to as a *coupling*.

For the purpose of this discussion, the assumption is that the links between the local and remote XIV Storage Systems are already established, as described in 4.11.2, “Remote mirror target configuration” on page 104.

6.1.1 Volume mirroring setup and activation

Volumes or consistency groups that participate in mirror operations are configured in pairs. These pairs are called peers. One peer is the source of the data to be replicated and the other is the target. The source has the role of master and is the controlling entity in the mirror. The target has the role of slave and it is normally controlled by operations performed by the master.

When initially configured, one volume is considered the source and is located at the primary system site and the other is the target and is located at the secondary system site. This designation is associated with the volume and its XIV Storage System and does not change. During various operations, the *role* can change between master and slave, but one system is always the primary and the other is always the secondary, for a pair.

Asynchronous mirroring is initiated at intervals that are defined by the *sync job schedule*. A sync job entails synchronization of data updates that are recorded on the master since the last successful synchronization. The sync job schedule is defined for both the primary and secondary system peers in the mirror. Having it defined for both entities enables an automated failover scenario, where the slave becomes a master and has a readily available schedule interval.

The system supports numerous schedule intervals ranging from 20 seconds to 12 hours. Consult an IBM representative to determine the optimum schedule interval based on your recovery point objective (RPO) requirements.

A schedule set to `NEVER` means that no sync jobs will be automatically scheduled. Thus issuing replication in that case must be done through an explicit manual command. For more information, see 6.6, “Detailed asynchronous mirroring process” on page 176. A manual command invocation can be done at any time, in addition to scheduled snapshots. These ad-hoc snapshots are issued from the master and trigger a sync job that is queued behind the outstanding sync jobs. See 6.5.4, “Mirrored snapshots” on page 172 for details.

The XIV Storage System GUI automatically creates schedules based on the RPO selected for the mirror being created. The interval can be set in the mirror properties panel or explicitly specified through the XCLI.

Note: Typical asynchronous mirror configuration indicates the RPO requirements, and the XIV Storage System automatically assigns an interval schedule that is one-third of that value (rounding down if needed).

Tip: The XIV Storage System allows a specific RPO and schedule interval to be set for each mirror coupling.

Also, be aware that existing slave volumes must be formatted before they are configured as part of a mirror. This means that the volume must not have any snapshots and must be unlocked. Otherwise, a new slave volume can be created, and used, when defining the mirror.

Use either the XIV Storage System GUI or the XCLI to create a mirror; both methods are illustrated in the following sections.

Using the GUI for volume mirror setup

From the XIV GUI, select the primary XIV Storage System and click the **Remote Function** icon, and select **Mirroring**, as shown in Figure 6-1.

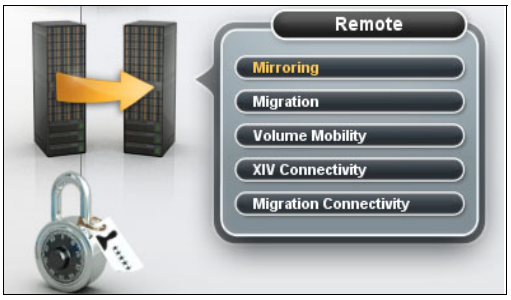


Figure 6-1 Selecting remote mirroring

To create a mirror, complete the following steps:

1. Click **Mirror Volume / CG**, as shown in Figure 6-2, and specify the source or master peer for the mirror pair, known as the coupling, and also other settings.

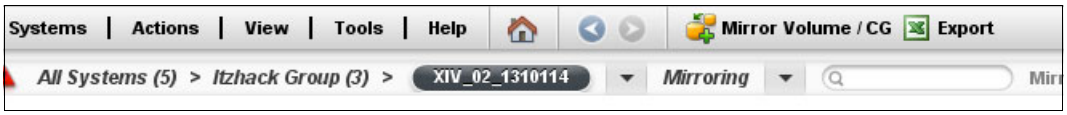


Figure 6-2 Selecting Create Mirror

There are other ways to create the couplings from the GUI. One way is to right-click a volume and select **Mirror Volume** from the Volumes and Snapshots list panel. Regardless of method used, the Create Mirror dialog box (Figure 6-3 on page 152) opens.

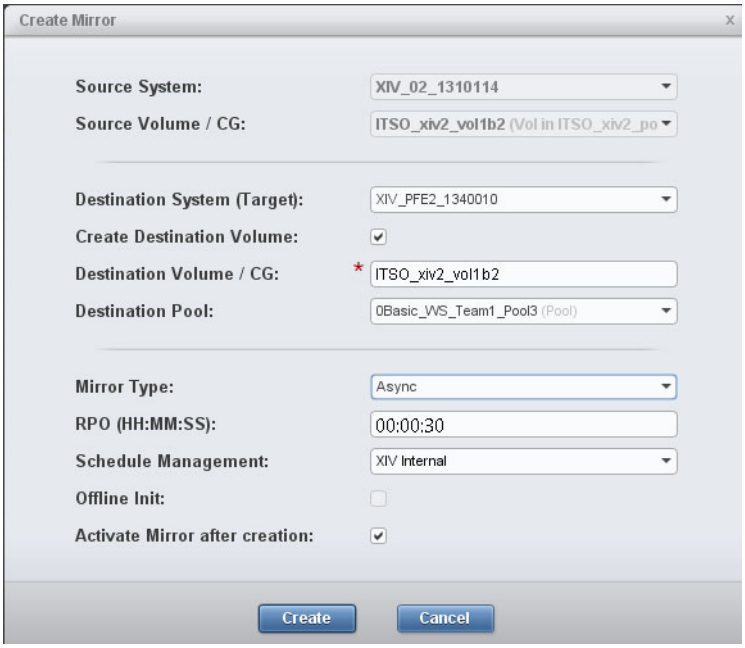


Figure 6-3 Create Async Mirror parameters

2. Make the selections you want for the coupling:
   - Source Volume / CG

     This is the volume or CG at the primary site to be mirrored. Select the volume or CG from the list. Consistency groups are shown in bold and are at the bottom of the list.
   - Destination System (Target)

     This is the XIV Storage System at the secondary site that will contain the target volumes, or slave peers. Select the secondary system from a list of known targets.
   - Create Destination Volume

     If selected, the destination volume will be created automatically. If unselected, the volume must be manually specified.

     Note: The slave volume must be unlocked and created as *formatted* (the default when created), which also means no associated snapshots.

     If the target volumes does not exist on the secondary XIV Storage System, check the **Create Destination** option to create an identically named and sized volume at the target site. In addition, a desired storage pool has to be selected in which the volume will be created. This pool must already exist at the remote site.
   - Destination Volume / CG

     This is the name of the destination volume, or CG, at the secondary site. If the **Create Destination** option was selected, the default is to use the same name as the source, but this can be changed. If the **Create Destination** option was not selected, the target volume or CG must be selected from the list.

     If the target volume already exists on the secondary XIV Storage System, the volume must be exactly the same size as the source; otherwise, a mirror cannot be set up. In this case, use the Resize function of the XIV Storage System to adjust the capacity of the target volume to match that of the source volume.

If you need to resize a source volume in an asynchronous mirroring relationship, you must first delete the mirror. You can then resize the source and target respectively and re-establish the mirror pair, using the Offline Init (trucking) feature.

– Destination Pool

  This is the storage pool on the secondary XIV Storage System that will contain the mirrored slave volumes. As stated before, this pool must already exist. This option is made available only if the **Create Destination** option is selected.

– Mirror Type

  In the Mirror Type field, change the selection from **Sync** to **Async**. (Sync is described in Chapter 5, “Synchronous Remote Mirroring” on page 113.)

– RPO

  The recovery point objective (RPO) time designation is the maximum time interval at which the mirrored volume or CG can lag behind the current or behind the source volume. The system strives to make a consistent copy of the destination CG or volume before the RPO is reached.

– Schedule Management

  Set the Schedule Management field to `XIV Internal` to create automatic synchronization using scheduled sync jobs. The `External` option specifies no sync jobs are scheduled by the system to run for this mirror and the interval will be set to `Never`. With this setting you will need to run an ad-hoc mirror snapshot to initiate a sync job.

– Offline Init

  This field is only available for selection if the **Create Destination** option is not selected. This engages the *trucking* feature of the XIV Storage System that enables initialization of the remote mirror slave peer without requiring the contents of the local source peer to be fully replicated over an inter-site link. See “Offline initialization” on page 156 for further information pertaining to offline initialization.

After the mirror is established, you can access the Mirror Properties panel by right-clicking the mirror; the interval can be changed here if necessary. Figure 6-4 on page 154 shows the panel and predefined interval schedules available depending on the defined RPO.

Important: The predefined interval schedule of *min_interval*, seen in the GUI, has a predefined interval value of 00:00:20 on the XIV Storage System.

Figure 6-4 Mirror Properties

The Mirroring panel shows the current status of the mirrors. Figure 6-5 and Figure 6-6 show the couplings created in standby, inactive mode on both the primary and secondary XIV Storage Systems respectively. With asynchronous mirrors, the RPO column is populated.

Figure 6-5 Coupling on the primary XIV Storage System in standby, inactive mode

Figure 6-6 Coupling on the secondary XIV Storage System in standby, inactive mode

Using XCLI for volume mirroring setup

Tip: When working with the XCLI session or the XCLI from a command prompt, the interface looks similar and commands can inadvertently be run to the incorrect XIV Storage System. Therefore, a good idea is to issue a **config_get** command to verify that the correct XIV Storage System is being addressed, or pay close attention to the command-line prompt that denotes the respective XIV (such as `XIV-02-1310114>>` as depicted next).

To set up the volume mirroring using XCL, follow these steps:

1. Open an XCLI session for the primary and the secondary XIV Storage System and run the identical **schedule_create** commands on each. Example 6-1 shows the syntax.

Example 6-1 Create schedules for remote mirroring

```
# On the primary
XIV-02-1310114>>schedule_create schedule=fifteen_min interval=00:15:00
Command executed successfully.
# Onthe secondary
XIV PFE-GEN3-131013>>schedule_create schedule=fifteen_min interval=00:15:00
Command executed successfully.
```

2. On the primary, run the **mirror_create** command shown in Example 6-2.

Example 6-2 Create remote mirror coupling

```
XIV-02-1310114>>mirror_create vol=async_test_2 create_slave=yes
remote_pool=ITSO slave_vol=async_test_2 type=async_interval target="XIV
PFE-GEN3-1310133" schedule=fifteen_min remote_schedule=fifteen_min rpo=3600
remote_rpo=3600
Command executed successfully.
```

3. To list the couplings on the primary XIV Storage System, run the **mirror_list** command used in Example 6-3. Note the status of `Initializing` is used in the XCLI when the coupling is in standby, inactive, or is initializing status.

Example 6-3 Listing mirror couplings on the primary

```
XIV-02-1310114>>mirror_list
Name           Mirror Type     Mirror Object Role   Remote System          Remote Peer     Active   Status      Link Up
async_test_1   async_interval  Volume        Master XIV PFE-GEN3-1310133   async_test_1    no       Initializing    yes
async_test_2   async_interval  Volume        Master XIV PFE-GEN3-1310133   async_test_2    no       Initializing    yes
```

4. To list the couplings on the secondary XIV Storage System, run the **mirror_list** command, as shown in Example 6-4. Note the status of `Initializing` is used when the coupling is in standby, inactive, or is initializing status.

Example 6-4 Listing mirror couplings on the primary

```
XIV PFE-GEN3-1310133>>mirror_list
Name           Mirror Type     Mirror Object Role   Remote System          Remote Peer     Active   Status      Link Up
async_test_1   async_interval  Volume        Slave  XIV-02-1310114         async_test_1    no       Initializing    yes
async_test_2   async_interval  Volume        Slave  XIV-02-1310114         async_test_2    no       Initializing    yes
```

5. Repeat steps 1 - 3 to create additional mirror couplings.

Offline initialization

Offline initialization is also referred to as *trucking*. It is a replication feature that provides the ability for a remote target volume to be initialized without requiring the contents of the source volume to be replicated over an inter-site link to accomplish the initialization phase of either of the XIV Storage System replications methods, namely synchronous and asynchronous. This is particularly helpful if the source volume already has a large amount of data that would normally be a lengthy process to transfer during normal initialization. Offline initialization can shorten the XIV Storage System mirror initialization phase significantly.

Offline initialization is accomplished with the following steps:

1. Create a snapshot of the future source volume on the primary XIV Storage System. The volume is a live production volume that is not currently in a mirroring relationship. Transferring the data on this snapshot to a volume on the secondary XIV Storage System is the objective of the offline initialization.
2. Map this snapshot to a host system and create a disk image of the volume. This image can be written to a file or to a tape or any other suitable media.

   Important: To create a valid disk image**,** use a backup tool that guarantees that the same data will be in the same location on both the source and backup copies on disk. A file-level backup tool will *not* work for that purpose; you need a backup tool that creates a raw copy, reading the entire disk serially, with no concept of files.

3. Transport this disk image to the XIV Storage System secondary. The volume can either be moved physically through carrier or electronically (FTP server).
4. Create the XIV Storage System volume on the XIV Storage System secondary that is exactly the same size as the source volume and map this volume to a host. This will be the future target volume, but it is not yet in any mirroring relationship.
5. Copy the disk image to the newly created volume on the XIV Storage System secondary using the same utility that was used to create the disk image.
6. Create the mirror relationship of your choice, being sure *not* to select the Create Destination option, and to explicitly name the destination volume based on the name of the volume created on the secondary site. And most important, select the **Offline Init** check box.
7. Activate the mirror relationship.

At this point, the XIV Storage System respective (either asynchronous or synchronous) mirror functions create the mirror pair. In the case of an asynchronous mirror, a most-recent snapshot is taken on the primary XIV Storage System. This most-recent snapshot on the primary is then compared to the data-populated target volume on the secondary XIV Storage System using 64 KB checksum exchanges. The new data that has been written to the source volume since the original snapshot was taken will be calculated. Only the new data on the source volume will be transferred to the target volume during the offline initialization phase.

XCLI command to create offline initialization asynchronous mirror

Creating an offline initialization asynchronous mirror is much like the XCLI process described earlier in the chapter, only now the `init_type` parameter must be used to specify offline, as shown in Example 6-5.

Example 6-5 XCLI to create offline initialization asynchronous mirror

```
XIV-02-1310114>>mirror_create vol=async_test_3 create_slave=yes remote_pool=ITSO
slave_vol=async_test_3 type=async_interval target="XIV PFE-GEN3-1310133"
schedule=fifteen_min remote_schedule=fifteen_min rpo=7200 remote_rpo=7200
init_type=offline
Command executed successfully.
```

Activating the remote mirror coupling using the GUI

To activate the mirrors, on the primary XIV Storage System go the Remote Mirroring menu and select the couplings to activate, right-click, and select **Activate**, as shown in Figure 6-7.

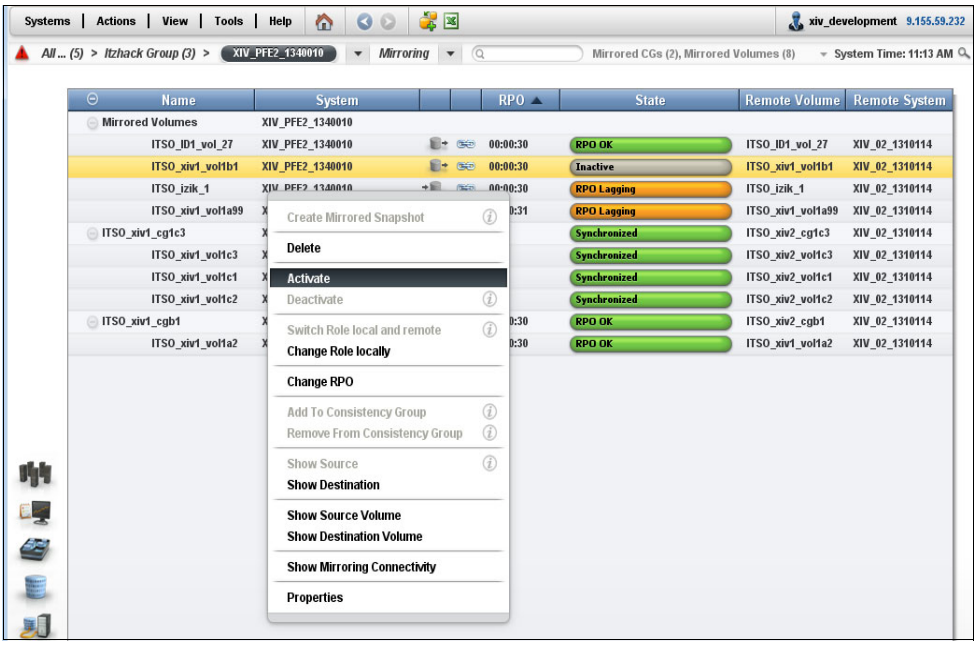


Figure 6-7 Activate mirror coupling

The `Initialization` status is shown just after the mirror coupling has been activated. As seen in Figure 6-8, after initialization is complete the Mirroring panel shows the status of the active mirrors as `RPO OK`.

Figure 6-8 Mirror status

Note: The mirroring status reported on the secondary XIV Storage System is `RPO OK`.

6.1.2 Consistency group configuration

IBM XIV Storage System leverages its consistency group capability to allow for mirroring related volumes at once. The system creates snapshots of the source consistency groups at scheduled intervals and synchronizes these point-in-time snapshots with the destination consistency groups.

Setting the consistency group to be mirrored is done by, first creating an empty consistency group, then pairing and synchronizing it with the consistency group on the peer. Next, each volume of the CGs (one by one) must be mirrored and reach RPO OK status. You can wait for all volumes to reach that status and add them all together to the CG, or add each volume to the CG when it reaches the RPO OK state. A consistency group must be created at the primary XIV Storage System and a corresponding consistency group at the secondary XIV Storage System. Creation of the consistency group entails specifying a name, which can be different for the two peers, and selecting storage pools, which can also have different names than those on the source side.

The GUI provides shortcuts to the tasks that must be done on the secondary system.

At the bottom of Figure 6-9 there are four smaller images of pop-up menus, which are the shortcuts to actions that you might need to do on the target system. The pop-up menus serve as a wizard and provide quick access to the peer system so you can do the tasks there. If the shortcut is used, the focus returns to the current context on the primary system after the action competes on the target.

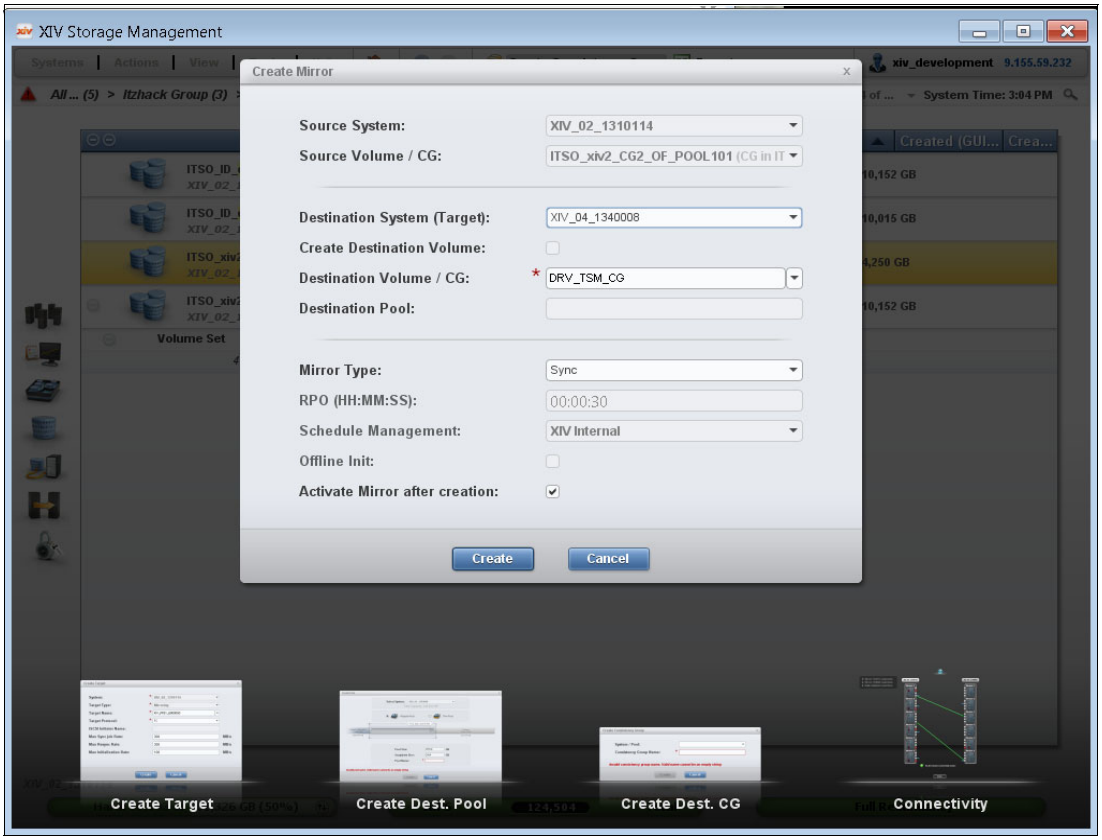


Figure 6-9 shortcuts to actions that might be needed on the target system

Important: All volumes to be added to a mirroring consistency group must be defined in the same pools at the primary. The same is true of the secondary site.

To create a mirrored consistency group, first create or select a CG on the primary and secondary XIV Storage Systems. Then, select the CG at the primary and specify **Mirror Consistency Group**.

Another way to create the consistency group mirror in the GUI is to select **Create Mirror** from the mirroring view. The Create Mirror dialog is shown in Figure 6-10.

Note: The consistency groups window is accessible by way of the Volumes function menu in the XIV Storage Management software.

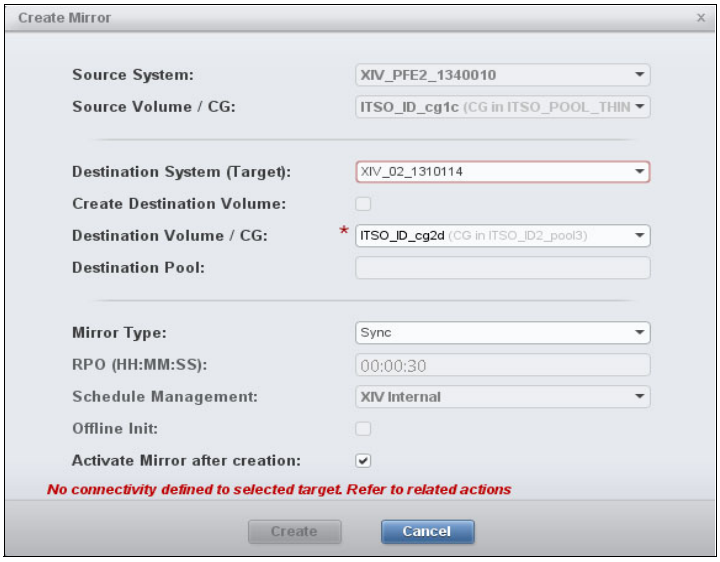


Figure 6-10 Asynchronous mirrored CG

Tip: Scroll to the bottom of the Source CG / Volume and respective Destination CG / Volume drop-down lists to select consistency groups presented to the user in bold text.

Volume pairs with different mirroring parameters will be automatically changed to match those of the CG when attempting to add them to the CG with the GUI.

Note: A consistency group that contains volumes cannot be mirrored unless all volumes are removed from the group, the group is mirrored, and then the volumes added again.

Adding a mirrored volume to a mirrored consistency group

The mirrored volume and the mirrored consistency group must have the following attributes:

- The volume is on the same system as the consistency group.
- The volume belongs to the same storage pool as the consistency group.

- The volume and consistency group are in RPO OK state.
- The volume and consistency group special snapshots, known as *last-replicated snapshots*, have identical timestamps. This means that the volumes must have the same schedule and at least one interval has passed since the creation of the mirrors.

  For more information about asynchronous mirroring special snapshots see 6.5.5, “Mirroring special snapshots” on page 174.

Also, mirrors for volumes must be activated before volumes can be added to a mirrored consistency group. This activation results in the initial copy being completed and scheduled sync jobs being run to create the special last-replicated snapshots.

Be careful when you add volumes to the mirrored CG because the RPO and schedule are changed to match the values set for the mirrored consistency group. It is possible that during the process the status changes or the last-replicated time stamp might not yet be updated. If an error occurs, verify the status and repeat the operation.

Go to the Mirroring panel and verify the status for the volumes to be added to the CG. Select each volume and click **Add To Consistency Group** (Figure 6-11).

Figure 6-11 Adding Mirrored Volumes to Mirror CG

Then, specify the mirrored consistency group, as shown in Figure 6-12.

Figure 6-12 Select Mirrored Consistency Group

Tip: A quick way to add eligible volumes to a CG is to select them using the CTRL key, and then drag and drop them into the CG mirror.

The Consistency Groups panel (Figure 6-13) shows the last-replicated snapshots. If the sync job is currently running, there will also be a most-recent snapshot that can be used, to restore the mirror to a consistent state, if the sync job fails midway because the data sent is not ordered, which can make the mirror inconsistent. The mirror is guaranteed to be consistent when all data is copied.

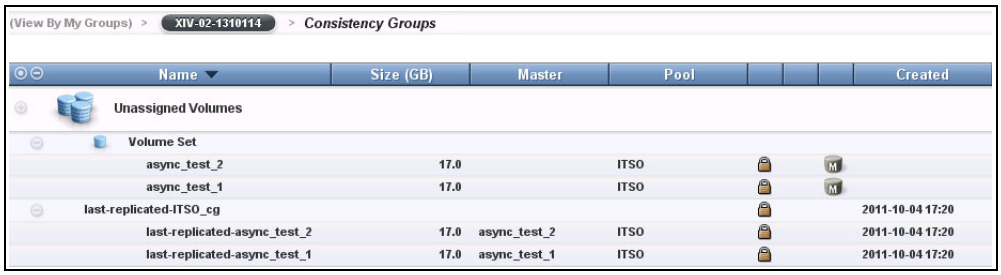


Figure 6-13 Mirrored CG, most-recent snapshot

Removing a volume from a mirrored consistency group

When removing a volume from a mirrored consistency group on the primary system, the corresponding peer volume will be removed from the peer consistency group. Mirroring is retained with the same configuration as the consistency group from which it was removed. All ongoing consistency group sync jobs keep running.

Note: The volume and CG must be in a status of `RPO OK` for removal of a volume from the group.

6.1.3 Coupling activation, deactivation, and deletion

Mirroring can be manually activated and deactivated per volume or CG pair. When it is activated, the mirror is in *active* mode. When it is deactivated, the mirror is in *inactive* mode. These modes have the following functions:

- Active

  Mirroring is functioning and the data is being written to the master and copied to the slave peers at regular scheduled intervals.

- Inactive

  Mirroring is deactivated. The data is not being replicated to the slave peer. The writes to the master continue. Upon reactivation, the last-replicated snapshot is consulted in order to transfer only the changes that were not copied since the deactivation. Inactive mode is used mainly when maintenance is performed on the secondary XIV system.

The mirror has the following characteristics:

- When a mirror is created, it is always initially in inactive mode.
- A mirror can only be deleted when it is in inactive mode.
- A consistency group can only be deleted if it does not contain any volumes associated with it.

- Transitions between active and inactive states can be performed only from the master XIV Storage System peer.
- In a situation where the primary XIV Storage System becomes unavailable, execution of a role change transitions the slave peers at the secondary site to a master peers role so that work can resume at the secondary. However, until the primary site is recovered, the role of its volumes cannot be changed from master to slave. In this case, both sides have the same role. When the primary site is recovered and before the link is resumed, first change the role from master to slave at the primary (see also 6.3, “Resynchronization after link failure” on page 168, and 6.4, “Disaster recovery” on page 169).

The mirroring is halted by deactivating the mirror and is required for the following actions:

- Terminating or deleting the mirroring
- Stopping the mirroring process:
  - For a planned network outage
  - To reduce network bandwidth
  - For a planned recovery test

The deactivation pauses a running sync job and no new sync jobs will be created if the active state of the mirroring is not restored. However, the deactivation does not cancel the status check by the master and the slave. The synchronization status of the deactivated mirror is calculated as though the mirror was active.

Change RPO and interval

The asynchronous-mirroring-required RPO can be changed as Figure 6-14 shows. For example, as Figure 6-15 shows, the RPO was changed to two hours (02:00:00) from one hour (01:00:00). The interval schedule can then be changed from the Properties panel as shown in Figure 6-16 on page 163.

Figure 6-14 Change RPO

(View By My Groups) > XIV-02-1310114 > Mirroring
Name
RPO
Status
Remote Volume
Remote System
Mirrored Volumes
async_test_a 01:00:00 RPO OK async_test_a XIV-05 G3 7820016
ITSO_cg 02:00:00 RPO OK ITSO_cg XIV-05 G3 7820016
async_test_1 02:00:00 RPO OK async_test_1 XIV-05 G3 7820016
async_test_2 02:00:00 RPO OK async_test_2 XIV-05 G3 7820016

Figure 6-15 New RPO value

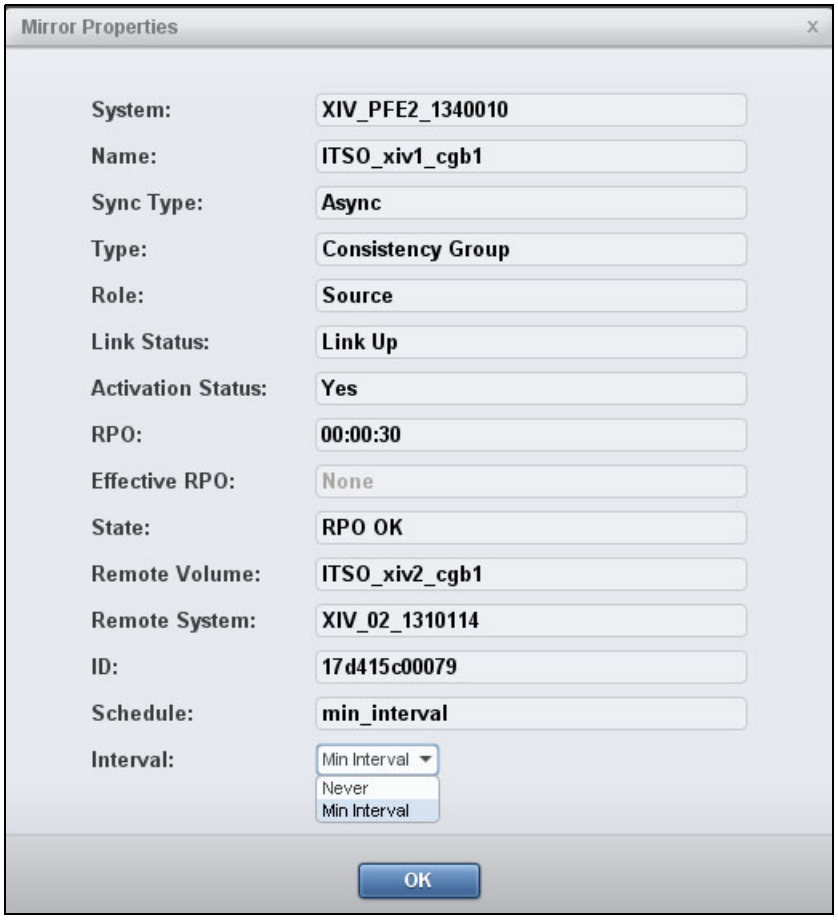


Figure 6-16 Change interval

Using XCLI commands to change RPO and schedule interval

Example 6-6 illustrates the use of XCLI commands to change the RPO and schedule interval.

Example 6-6 XCLI commands for changing RPO and schedule interval

```
XIV-02-1310114>>mirror_change_rpo cg=ITSO_cg rpo=7200 remote_rpo=7200
Command executed successfully.
XIV-02-1310114>>schedule_change schedule=thirty_min interval=00:30:00 -y
Command executed successfully.
XIV-02-1310114>>mirror_change_remote_schedule cg=ITSO_cg
remote_schedule=thirty_min
Command executed successfully.
```

Note: In Example 6-6, the schedule must first be created on the remote, secondary XIV Storage System, prior to running the **`mirror_change_remote_schedule`** XCLI command.

Deactivation on the master

To deactivate a mirror, right-click the desired mirror and select **Deactivate**, as shown in Figure 6-17

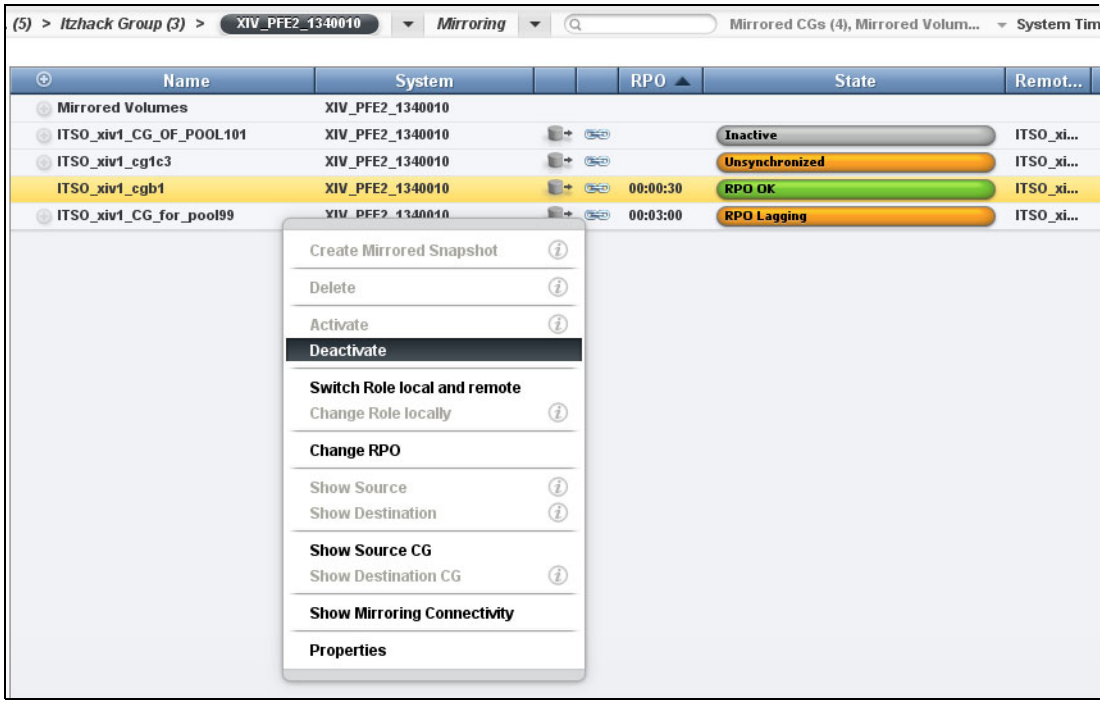


Figure 6-17 Mirror deactivate

The activation state changes to inactive and subsequently replication pauses. Upon activation, the replication resumes.

An ongoing sync job resumes upon activation. No new sync job will be created until the next interval.

Deactivation on the slave

Deactivation on the slave is not available, regardless of the state of the mirror. However, the peer role can be changed to master, which sets the status to inactive.

For consistency group mirroring, deactivation pauses all running sync jobs pertaining to the consistency group.

Using XCLI commands for deactivation and activation

Example 6-7 shows XCLI commands for CG deactivation and activation.

Example 6-7 XCLI commands for CG deactivation and activation

```
# Deactivate
XIV-02-1310114>>mirror_deactivate cg=ITSO_cg -y
Command executed successfully.
# Activate
XIV-02-1310114>>mirror_activate cg=ITSO_cg
Command executed successfully.
```

Mirror deletion

A mirror relationship can be deleted only when the mirror pair (volume pairs or a consistency group) is inactive. When the mirror is deleted, the associated information regarding the relationship is removed. If the mirror must be re-established, the XIV Storage System must again do an initial copy from the source to the destination volume.

When the mirror is part of a consistency group, volume mirrors must first be removed from the mirrored CG. For a CG, the last-replicated snapgroup for the master and the slave CG must be *deleted* or *disbanded* (making all snapshots directly accessible) after deactivation and mirror deletion. This CG snapgroup is recreated with only the current volumes after the next interval completes. The last-replicated snapshots for the mirror can now be deleted, allowing a new mirror to be created. All existing volumes in the CG need to be removed before the CG can be deleted. It is possible to delete an inactive CG with all of its mirrors in one action.

When the mirror is deleted, the slave volume becomes a normal volume again, but the volume is *locked,* which means that it is write protected. To enable writing to the volume go to the Volumes list panel, select the volume by right-clicking it, and select **Unlock**.

Note: The slave volume must also be formatted before it can be part of a new mirror unless, offline initialization is selected. Formatting a volume requires all of its snapshots to be deleted.

6.2 Role reversal

Mirroring roles can be modified by either *switching* or *changing* roles.

- *Switching* roles must be initiated on the master volume/CG when the remote mirroring is operational. As the task name implies, it switches the master role to the slave role and at the same time, at the secondary site, switches the slave role to the master role.
- *Changing* roles can be performed at any time, whether the pair is active or inactive. The source can be changed also, when the mirror is inactive. A change role reverts only the role of the addressed peer. The *switch roles* command is only available on the source peer when both the source and destination XIV systems are accessible. The direction of the mirror can be reversed also, by following a process of multiple *change role* operations.

6.2.1 Switching roles

Switch roles is a useful command when performing a planned site switch by reversing the replication direction. It is available only when both the master and slave XIV Storage Systems are accessible. Mirroring must be active and synchronized, and RPO OK, to issue the command.

Attention: Because the slave system might be up to the RPO interval behind the master, an indiscriminate use of “switch roles” can result in the master being overwritten with data that is up to the RPO interval older than the master, thus resulting in the loss of data. Switch roles must only be used in the case when there has been zero host I/O since the last sync job was run, as would be the case in switching from a DR site back to a master, where the DR host is idled, a sync job is run, and then roles are switched.

The command to switch roles can be issued only for a master volume or CG (Figure 6-18).

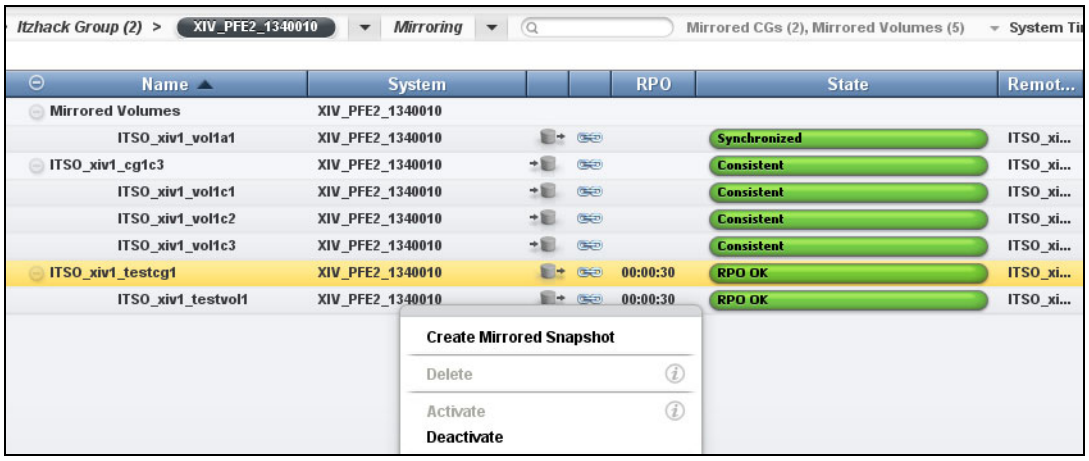


Figure 6-18 Switch roles of a master consistency group

A confirmation window opens (Figure 6-19) so you can confirm switching roles.

Figure 6-19 Verify switch roles

Normally, switching the roles requires shutting down the applications at the primary site first, changing the SAN zoning and XIV LUN masking to allow access to the secondary site volumes, and then restarting the application with access to the secondary XIV Storage System. Thus, role-switching is only one step in the process and is not the sole reason for the work disruption.

6.2.2 Change role

In the event of a disaster at the primary site, a role change at the secondary site is the normal recovery action.

Assuming that the primary site is down and that the secondary site must now become the main production site, changing roles is performed at the secondary (new production) site, so that production can be resumed using the newly appointed primary storage. Later, when the primary site is up again and communication is re-established, perform a change role at the primary site to set the previous master to slave, which facilitates the mirroring from the secondary site (which became primary) back to the primary (which became secondary) site. That way the data is kept in sync on both sites and, no data is lost. This completes a *switch role* operation.

Note: After the data has been synchronized from the secondary site to the primary site, a switching roles can be performed to again make the primary site the master.

Changing the slave peer role

The role of the slave volume or consistency group can be changed to the master role, as shown in Figure 6-20.

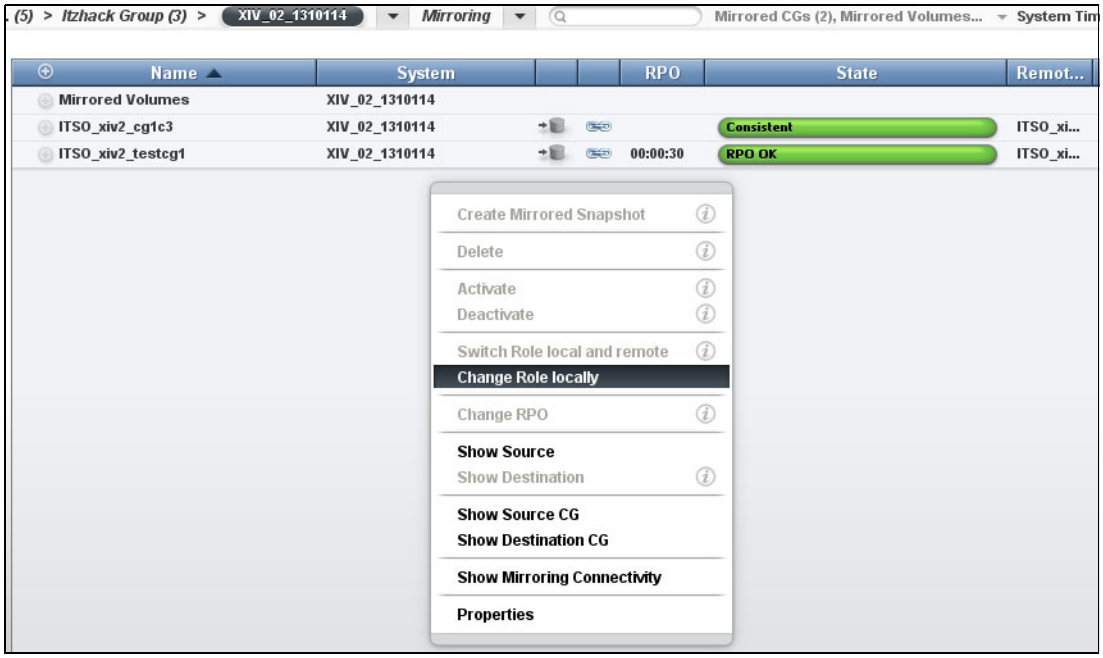


Figure 6-20 Change role of a slave mirrored volume

As shown in Figure 6-21, you are then prompted to confirm the role change, a role reversal.

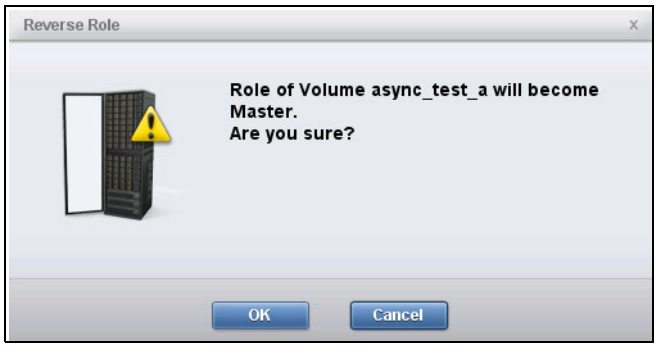


Figure 6-21 Verify change role

After this changeover, the following situation is true:

- The slave volume or consistency group is now the master.
- The last-replicated snapshot is restored to the volumes.
- The coupling remains in inactive mode. This means that remote mirroring is deactivated. This ensures an orderly activation when the role of the peer on the other site is changed.

The new master volume or consistency group starts to accept write commands from local hosts. Because coupling is not active changes are being tracked on the source XIV system.

After changing the slave to the master, an administrator must also change the original master to the slave role before mirroring can be activated. If both peers are kept in the same role of master, mirroring cannot be restarted.

Slave peer consistency

When the user is changing the slave volume or consistency group to a master volume or master consistency group, they might not be in a consistent state. Therefore, the volumes are automatically restored to the last-replicated snapshot. That means some data might be lost. The data that was written to the source and was not yet replicated is the one that will be lost. An application-level recovery action is necessary to reapply the lost writes.

Changing the master peer role

When a peer role is changed from slave to master, then the mirror automatically becomes inactive because both volumes are a master. When coupling is inactive, the master volume or consistency group can change roles. After such a change the master volume or consistency group becomes the slave volume or consistency group.

Unsynchronized master becoming a slave volume or consistency group

When a master volume or consistency group is inactive, it is also not consistent with the previous slave. Any changes made after the last-replicated snapshot time will be lost when the volume/CG becomes a slave volume/CG as the data is restored to the most recent available consistent data reflected by the last-replicated snapshot.

Upon re-establishing the connection, the primary volume or consistency group (the current slave volume/CG) is updated from the secondary volume/CG (which is now the new master volume/CG) with data that was written to the secondary volume after the last-replicated snapshot time stamp.

Reconnection when both sides have the same role

Situations where both sides are configured to the same role can only occur when one side was changed. The roles must be changed to have one master and one slave volume or consistency group. Change the volume roles as appropriate on both sides before the link is resumed.

If the link is resumed and both sides have the same role, the coupling does not become operational. The user must use the *change role* function on one of the volumes and then activate the mirroring.

Peer reverts to the last-replicated snapshot. See 6.5.5, “Mirroring special snapshots” on page 174.

6.3 Resynchronization after link failure

When a link failure occurs, the primary system must start tracking changes to the mirror source volumes so that these changes can be copied to the secondary after it is recovered.

When recovering from a link failure, the following steps are taken to synchronize the data:

- Asynchronous mirroring sync jobs proceed as scheduled. Sync jobs are restarted and a new most-recent snapshot is taken. See 6.5.5, “Mirroring special snapshots” on page 174.
- The primary system copies the changed data to the secondary volume. Depending on how much data must be copied, this operation could take a long time, and the status remains RPO_Lagging.

Also, if mirroring was suspended for disaster recovery tests at the secondary site, take measures to reset the changes made to the secondary site during the tests and before re-establishing mirroring from the primary to the secondary. The reason for this action is that the changes made on the secondary are not tracked and, if left intact, the data on the secondary will be inconsistent after the replication is resumed. To recover from an inconsistent secondary, the incumbent mirror must be deleted and a new one created.

If a disaster occurred and production is now running on the secondary site, re-establish mirroring first from the secondary site to the primary site. Later, switch mirroring to the original direction from the primary XIV Storage System to the secondary XIV Storage System.

6.3.1 Last-replicated snapshot

The last-replicated snapshot ensures that there is a consistent copy of the data for recovery purposes. This means either in the event the secondary site must be used to carry on production or resynchronization must be done after the master stopped suddenly, leaving the secondary copy in an inconsistent state (because it was only partially updated with the changes that were made to the master volume). This last-replicated snapshot is preserved until a volume/CG pair is synchronized again through a completed sync job, at which point a new last-replicated snapshot is taken.

The last-replicated snapshot lifecycle is as follows:

- A snapshot of the master volume or CG is taken at the primary, and is named the most-recent snapshot.
- All data from the most-recent snapshot is sent to the secondary site through a sync job.
- Upon the completion of the sync job, the secondary site takes its last-replicated snapshot and time-stamps it accordingly.
- Notification is sent from the secondary site to the primary site, and the most-recent snapshot is renamed to last-replicated snapshot.

6.4 Disaster recovery

There are two broad categories of disaster:

- One that destroys the primary site or destroys the data there
- One that makes the primary site or the data there unavailable, but leaves the data intact

However, within these broad categories several situations might exist. Among the disaster situations and recovery procedures are the following items:

- A disaster that renders the XIV unavailable at the primary site, yet the servers there are still available.

  In this scenario the volumes/CG on the XIV Storage System at the secondary site can be switched to master volumes/CG and servers at the primary site can be redirected to the XIV Storage System at the secondary site, and normal operations can be resumed. When the XIV Storage System at the primary site is recovered, the data can be mirrored from the secondary site back to the primary site. A full initialization of the data is usually not needed. Only changes that took place at the secondary site are transferred to the primary site. If you want, a planned site switch can then take place to resume production activities at the primary site. See 6.2, “Role reversal” on page 165, for details related to this process.

- A disaster that makes both the primary site XIV and servers unavailable.

  In this scenario the standby, inactive servers at the secondary site are activated and attached to the secondary XIV Storage System to allow for normal operations to start. This requires changing the role of the slave peers to become master peers.

  After the primary site is recovered, the data at the secondary site can be mirrored back to the primary site. This most likely requires a full initialization of the primary site because the local volumes might not contain any data. See 6.1, “Asynchronous mirroring configuration” on page 150, for details related to this process.

  When initialization completes the peer roles can be switched back to master at the primary site and the slave at secondary site. The servers are then redirected back to the primary site. See 6.2, “Role reversal” on page 165, for details related to this process.

- A disaster that breaks all links between the two sites but both sites remain running

  In this scenario the primary site continues to operate as normal. When the links are re-established the data at the primary site is resynchronized with the secondary site. Only the changes since the previous last-replicated snapshot are sent to the secondary site.

6.5 Mirroring process

This section explains the overall asynchronous mirroring process, from initialization to ongoing operations. The asynchronous mirroring process generates scheduled snapshots of the master peer at user-configured intervals and synchronizes these consistent snapshots with the slave peer (see “Snapshot lifecycle” on page 175). The secondary peer is not consistent throughout the actual copy process. When the snapshot copy is complete, the secondary is consistent again.

6.5.1 Initialization process

The mirroring process starts with an initialization phase:

1. A sync job is started from the source to the destination. The copy is done at the configured speed of max_initialization_rate. The speed of the initial copy is regulated so that it will not affect the production. At the initialization stage, no snapshots are involved. At the end of the initialization the secondary copy is not necessarily consistent, because writes orders were not kept throughout the process. When the initialization phase is complete, snapshots are taken on the source and destination respectively and used for the ongoing asynchronous replication.

2. The most-recent data is copied to the slave and a last-replicated snapshot of the slave is taken (Figure 6-22).

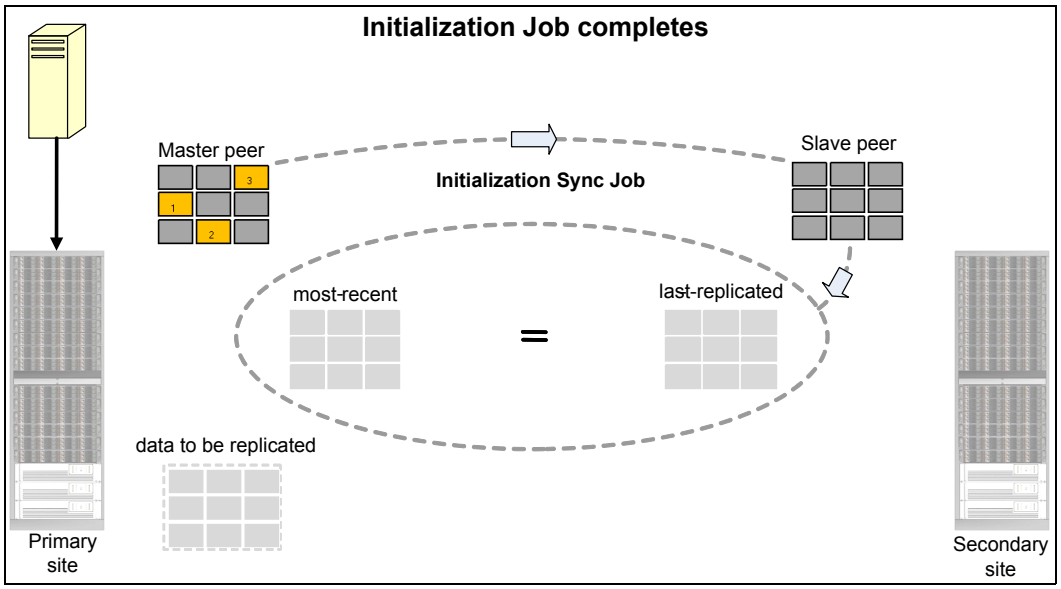


Figure 6-22 Initialization process completes

3. The most-recent snapshot on the master is renamed to last-replicated. This snapshot is identical to the data in the last-replicated snapshot on the slave (Figure 6-23).

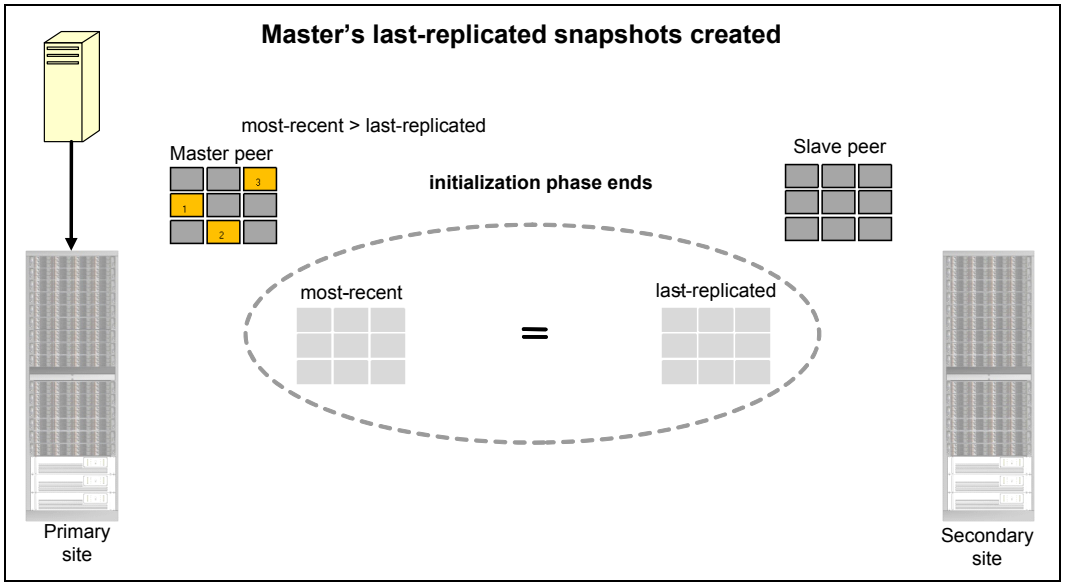


Figure 6-23 Ready for ongoing operation

4. Scheduled sync jobs are now able to run to create periodic consistent copies of the master volumes or consistency groups on the slave system. See 6.6, “Detailed asynchronous mirroring process” on page 176.

In the case of an offline initialization, the process proceeds as described previously, but upon activation, the two XIV Storage Systems exchange bitmaps to identify which partitions or blocks actually contain data, and then those blocks are checksummed and compared, and only blocks verified to differ are transferred from the master to the slave.

6.5.2 Ongoing mirroring operation

Following the completion of the initialization phase, the master examines the synchronization status at scheduled intervals and determines the scope of the synchronization. The following process occurs when a synchronization is started:

1. A snapshot of the master is created.
2. The master calculates the differences between the master snapshot and the most-recent master snapshot that is synchronized with the slave.
3. The master establishes a synchronization process called a sync job that replicates the differences from the master to the slave. Only data differences are replicated.

Details of this process are in 6.6, “Detailed asynchronous mirroring process” on page 176.

6.5.3 Mirroring consistency groups

The synchronization status of the consistency group is determined by the status of all volumes pertaining to this consistency group.

- The activation and deactivation of a consistency group affects all of its volumes.
- Role updates concerning a consistency group affect all of its volumes.
- It is impossible to directly activate, deactivate, or update the role of a given volume within a consistency group.
- It is not possible to directly change the schedule of an individual volume within a consistency group.

6.5.4 Mirrored snapshots

In addition to using the asynchronous, schedule-based option, mirrored snapshots (also called ad-hoc sync jobs) can be used. A user can manually create snapshots (while on the primary side only) of the coupling peers at both the local and remote sites, so that the local and remote snapshots are identical. Managed by the master peer, these snapshots can be issued regardless of whether the mirror pairing has a schedule, The action enqueues a sync job that is added behind the outstanding scheduled sync jobs and creates an ad hoc snapshot on the master and then on the slave.

Plausible use cases for ad-hoc snapshots are as follows:

- Accommodates a need for adding manual replication points to a scheduled replication process.
- Can be used to create application-consistent replicas (when the I/O of applications is paused or stopped), and the scheduled replication is not readily available. These snapshots are useful for backup and disaster recovery purposes.

The following characteristics apply to the manual initiation of the asynchronous mirroring process:

- Multiple mirror snapshot commands can be issued – there is no maximum limit, aside from space limitations.
- An active ad-hoc mirror snapshot delays the next scheduled interval-based snapshot, but does not cancel the creation of this sync job.
- The interval-based mirror snapshot will be cancelled only if the ad-hoc mirror snapshot is never completed.

Other than these differences, the manually initiated sync job is identical to a regular interval-based sync job.

Note: After the snapshots are created, they can be managed independently. This means the local mirror snapshot can be deleted without affecting its remote peer snapshot.

Use case scenarios for mirrored snapshots

A typical use case scenario for a mirrored snapshot is when the user has a production volume that is replicated to a remote site, and the user wants to create a snapshot at the remote that is application consistent. The user most likely does this for the any of the following reasons:

- To perform a disaster recovery test at the remote site using application consistent data.
- To create a clone of the production system at the remote site that is application consistent.
- To create a backup of the production system at the remote site that is application consistent. This option would allow the user to not have to back up hardware, such as a tape library, at the production site.

Complete the following steps to achieve this:

1. Ensure the mirror is established and working correctly. If the mirror is synchronous, the status should show as *Synchronized* at the production site.

   If the mirror is asynchronous it shows as *RPO OK*. If the mirror has a status of `RPO Lagging`, this indicates the link is already having problems mirroring the regular scheduled snapshots without adding an additional sync job to the list. Figure 6-24 shows the desired status on the primary site for creating mirrored snapshots.

Figure 6-24 Mirrored snapshot status

2. At the production site place the application into backup mode. This does not mean stopping the application, but instead means having the application flush its buffers to disk so that a hardware snapshot will contain application consistent data. This can momentarily cause poor performance.
3. On the production XIV Storage System, select the **Create Mirrored Snapshot** command as seen in Figure 6-25 on page 174.
4. Take the production site application out of backup mode.
5. On the remote XIV Storage System, confirm the creation of the new ad hoc snapshot. For synchronous mirrors this snapshot should be available immediately. For asynchronous mirrors, there might be a delay. This is because if a sync job is already running, the mirrored snapshot sync job must wait for it to complete. When the mirrored snapshot sync job completes, the snapshot at the remote site is available.

Figure 6-25 Create mirrored snapshot

6. On the remote XIV Storage System, unlock the new snapshot and map it to your host at the backup site.
7. Now using the remote site host you can perform application cloning, disaster recovery testing, or production site backups, all using application-consistent data.

XCLI commands for ad-hoc snapshots

Example 6-8 illustrates some XCLI commands for ad hoc snapshots.

Example 6-8 XCLI mirrored snapshot commands

```
#Create ad-hoc snapshot
XIV-02-1310114>>mirror_create_snapshot cg=ITSO_cg name=ITSO_cg.mirror_snapshot_1
slave_name=ITSO_cg.mirror_snapshot_1
Command executed successfully.

# List current and pending sync jobs
XIV-02-1310114>>sync_job_list
No pending jobs exist

# Cancel all mirrored snapshots; ad hoc sync jobs
XIV-02-1310114>>mirror_cancel_snapshot cg=ITSO_cg -y
Command executed successfully.
```

6.5.5 Mirroring special snapshots

The status of the synchronization process and the scope of the sync job are determined through the use of the following two special snapshots:

- Most-recent snapshot

  This snapshot is the most recent taken of the master system, either a volume or consistency group. This snapshot is taken prior to the creation of a new sync job. This entity is maintained on the master system only.

- Last-replicated snapshot

  This is the most-recent snapshot that has been fully synchronized with the slave system. This snapshot is duplicated from the most-recent snapshot after the sync job is complete. This entity is maintained on both the master and the slave systems.

Snapshot lifecycle

Throughout the sync job lifecycle, the most-recent and last-replicated snapshots are created and deleted to denote the completion of significant mirroring stages.

This mechanism has the following characteristics and limitations:

- The last-replicated snapshots have two available time stamps:
  - On the master system: the time that the last-replicated snapshot is copied from the most-recent snapshot
  - On the slave system: the time that the last-replicated snapshot is copied from the master system
- No snapshot is created during the initialization phase.
- Snapshots are deleted only after newer snapshots are created.
- A failure in creating a last-replicated snapshot caused by space depletion is handled in a designated process. See 6.8, “Pool space depletion” on page 184, for additional information.
- Ad hoc sync job snapshots that are created by the Create Mirrored Snapshot operation are identical to the last-replicated snapshot until a new sync job runs.

Table 6-1 indicates which snapshot is created for a given sync job phase.

Table 6-1 Snapshots and sync job phases

	Sync job phase	Most-recent snapshot	Last-replicated snapshot	Details
1	New interval starts.	Created on the master system.		The most-recent snapshot is created only if there is no sync job running.
2	Calculate the differences.			The difference between the most-recent snapshot and the last-replicated snapshot is transferred from the master system to the slave system.
3	The sync job is complete.		Created on the slave system.	The last-replicated snapshot on the slave system is created from the snapshot that has just been mirrored.
4	Following the creation of the last-replicated snapshot.		Created on the master system.	The last-replicated snapshot on the master system is created from the most-recent snapshot.

6.6 Detailed asynchronous mirroring process

After initialization is complete, sync job schedules become active (unless schedule=never or Type=external is specified for the mirror). This starts a specific process that replicates a consistent set of data from the master to the slave. This process uses special snapshots to preserve the state of the master and slave during the synchronization process. This allows the changed data to be quantified and provides synchronous data points that can be used for disaster recovery. See 6.5.5, “Mirroring special snapshots” on page 174.

The sync job runs and the mirror status is maintained at the master system. If a previous sync job is running, a new sync job will not start. The following actions are taken at the beginning of each interval:

1. The most-recent snapshot is taken of the volume or consistency group:
   a. Host I/O is quiesced.
   b. The snapshot is taken to provide a consistent set of data to be replicated.
   c. Host I/O resumes.
2. The changed data is copied to the slave:
   a. The difference between the most-recent and last-replicated snapshots is determined.
   b. This changed data is replicated to the slave.

   This step is illustrated in Figure 6-26.

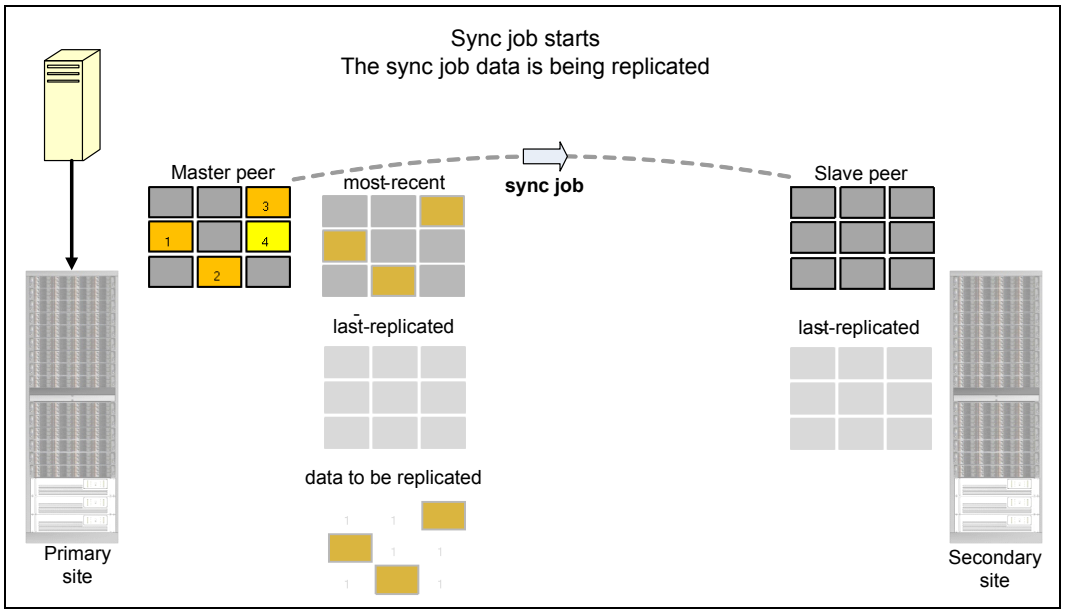


Figure 6-26 Sync job starts

3. A new last-replicated snapshot is created on the slave. This snapshot preserves the consistent data for later recovery actions if needed (Figure 6-27).

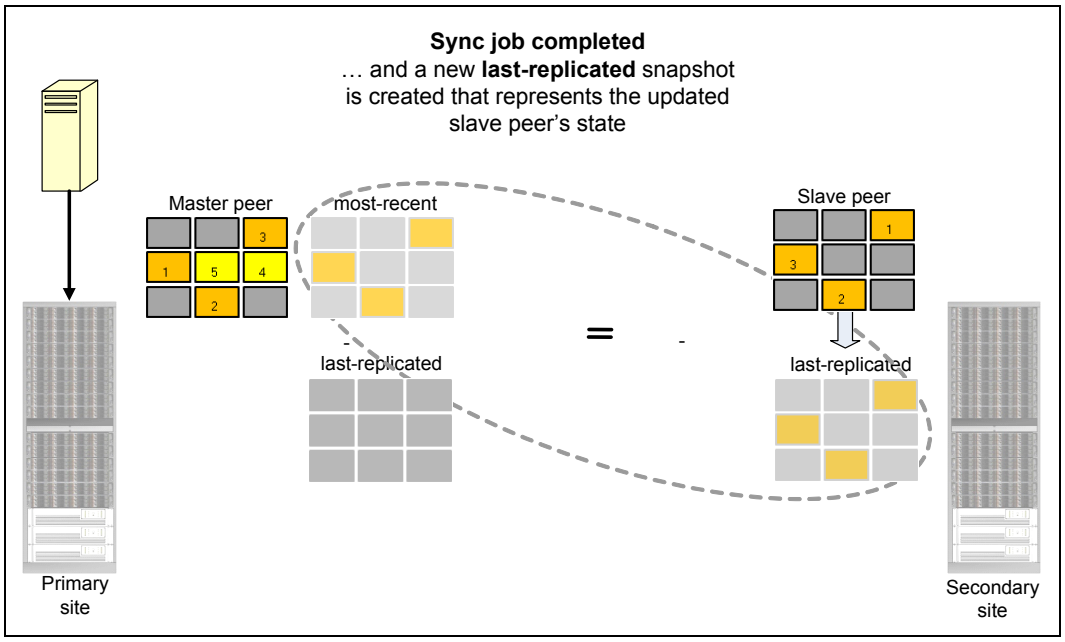


Figure 6-27 Sync job completes

4. The most-recent snapshot is renamed on the master (Figure 6-28):
   a. The most recent data is now equivalent to the data on the slave.
   b. Previous snapshots are deleted.
   c. The most-recent snapshot is renamed to last-replicated.

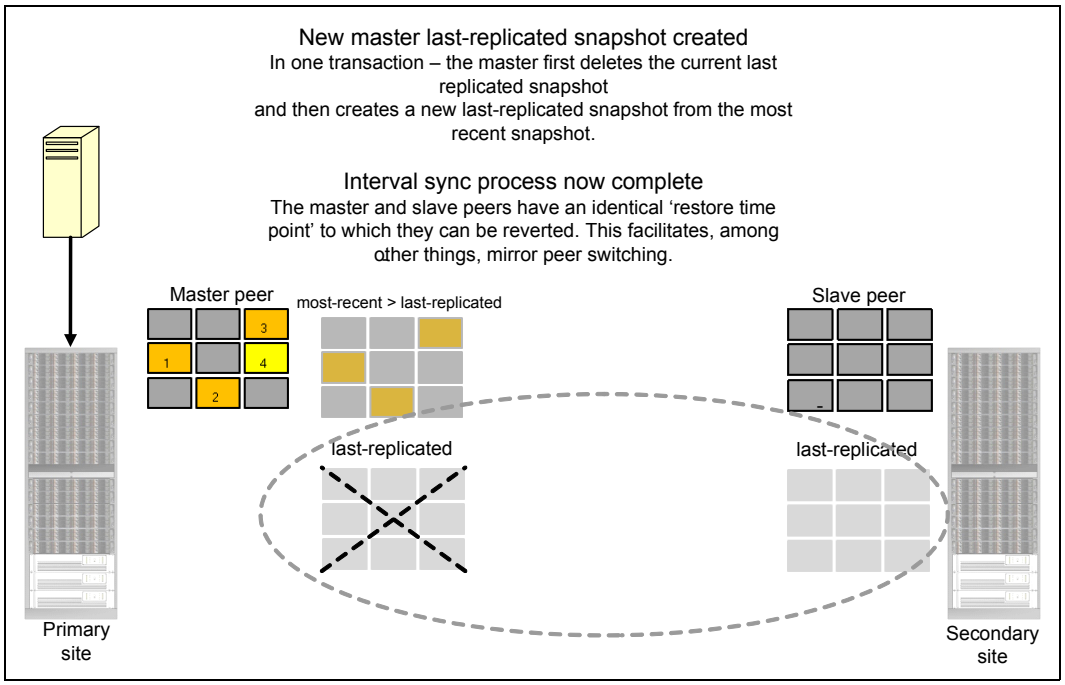

New master last-replicated snapshot created
In one transaction – the master first deletes the current last
replicated snapshot
and then creates a new last-replicated snapshot from the most
recent snapshot.
Interval sync process now complete
The master and slave peers have an identical 'restore time
point' to which they can be reverted. This facilitates, among
other things, mirror peer switching.
Master peer
most-recent > last-replicated
Slave peer
last-replicated
last-replicated
Primary
site
Secondary
site

Figure 6-28 New master's last-replicated snapshot

The next sync job can now be run at the next defined interval.

Mirror synchronization status

Synchronization status is checked periodically and is independent of the mirroring process of scheduling sync jobs. See Figure 6-29 for a view of the synchronization states.

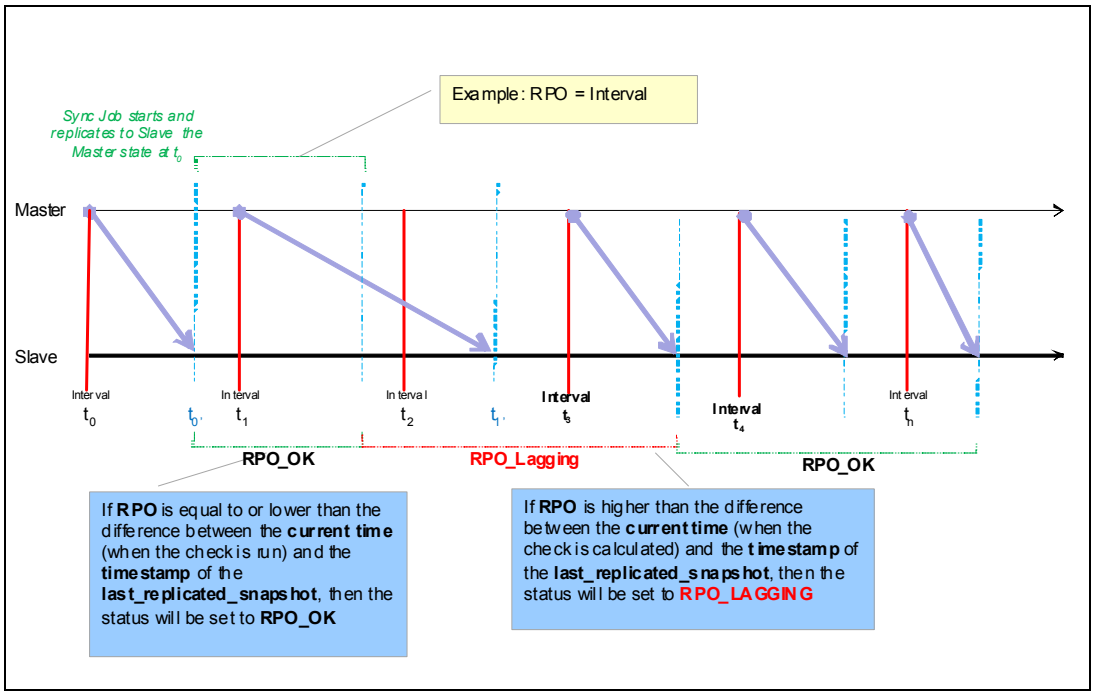

Example: RPO = Interval
Sync Job starts and replicates to Slave the Master state at t_0
Master
Slave
Interval t_0
Interval t_1
Interval t_2
Interval t_3
Interval t_4
Interval t_n
t_0'
t_1'
RPO_OK
RPO_Lagging
RPO_OK
If **RPO** is equal to or lower than the difference between the **current time** (when the check is run) and the **timestamp** of the **last_replicated_snapshot**, then the status will be set to **RPO_OK**
If **RPO** is higher than the difference between the **current time** (when the check is calculated) and the **timestamp** of the **last_replicated_snapshot**, then the status will be set to **RPO_LAGGING**

Figure 6-29 Synchronization states

The following synchronization states are possible:

- Initialization

  Synchronization does not start until the initialization completes.
- RPO_OK

  Synchronization has completed within the specified sync job interval time, RPO.
- RPO_Lagging

  Synchronization has completed but took longer than the specified interval time, RPO.

6.7 Asynchronous mirror step-by-step illustration

In the previous sections, the steps taken to set up, synchronize, and remove mirroring, utilizing both the GUI and the XCLI were explained. This section provides an asynchronous mirror step-by-step illustration.

6.7.1 Mirror initialization

This section is a continuation of the setup illustrated in 6.1, “Asynchronous mirroring configuration” on page 150, which assumes that the Fibre Channel ports are properly defined as source and targets, and all the physical paths are in place.

Mirrored volumes have been placed into a mirrored consistency group and the mirror has been initialized and has a status of RPO OK. See Figure 6-30.

Figure 6-30 Master status after setup

6.7.2 Remote backup scenario

One possible scenario related to the secondary site is to provide a consistent copy of data that is used as a periodic backup. This backup copy could be copied to tape or used for data-mining activities that do not require the most current data.

In addition to mirrored snapshots (see 6.5.4, “Mirrored snapshots” on page 172), the backup can be accomplished by creating a duplicate of the last-replicated snapshot of the slave consistency group at the secondary XIV Storage System. This new snapshot can then be mounted to hosts and backed up to tape or used for other purposes.

GUI steps to duplicate a snapshot group

From the Consistency Groups panel select **Duplicate** at the secondary site XIV Storage System, as shown in Figure 6-31.

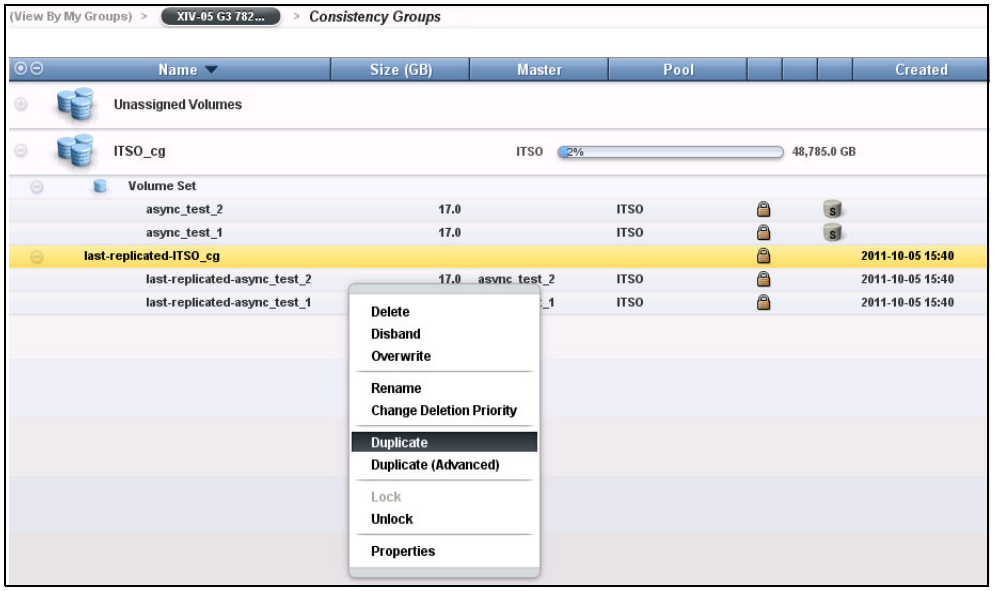


Figure 6-31 Duplicate last-replicated snapshot

A new snapshot group is created with the same time stamp as the last-replicated snapshot, as shown in Figure 6-32.

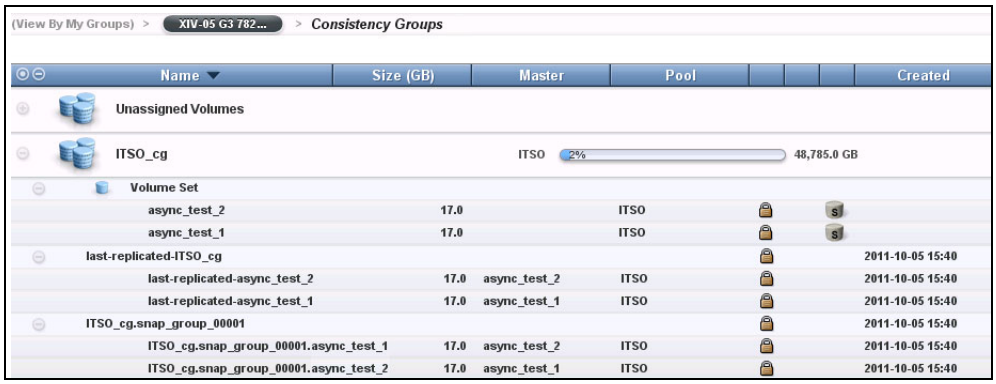


Figure 6-32 Duplicate snapshot

XCLI command to duplicate a snapshot group

Example 6-9 illustrates the **snap_group_duplicate** command at the secondary site XIV Storage System.

Example 6-9 XCLI to duplicate a snapshot group

```
XIV-05 G3 7820016>>snap_group_duplicate snap_group=last-replicated-ITSO_cg
new_snap_group=ITSO_cg.snap_group_00001
Command executed successfully.
```

6.7.3 DR testing scenario

It is important to verify disaster recovery procedures. This can be accomplished using the remote volumes with hosts at the recovery site to verify that the data is consistent and that no data is missing because of volumes not being mirrored. This process is partly related to making slave volumes available to the hosts, but it also includes processes external to the XIV Storage System commands. For example, the software available on the remote hosts and user access to those hosts must also be verified. The example presented only covers the XIV Storage System commands.

GUI steps for DR testing

There are two ways to verify the integrity of data on the slave volumes. The last-replicated snapshots can be duplicated, and those snapshots presented to the appropriate DR host and tested, or the slave system can be promoted to be a master, and a test with the actual volumes can be performed. The former has the advantage that the DR process is not suspended for the duration of the test, and the latter has the advantage of being the same process as an actual DR event. In actual production, a combination of the two test methodologies will likely be the best overall.

To promote the slave volumes the process begins by changing the role of the slave volumes to master volumes. This results in the mirror being deactivated. The remote hosts can now access the remote volumes. See Figure 6-33, Figure 6-34, and Figure 6-35.

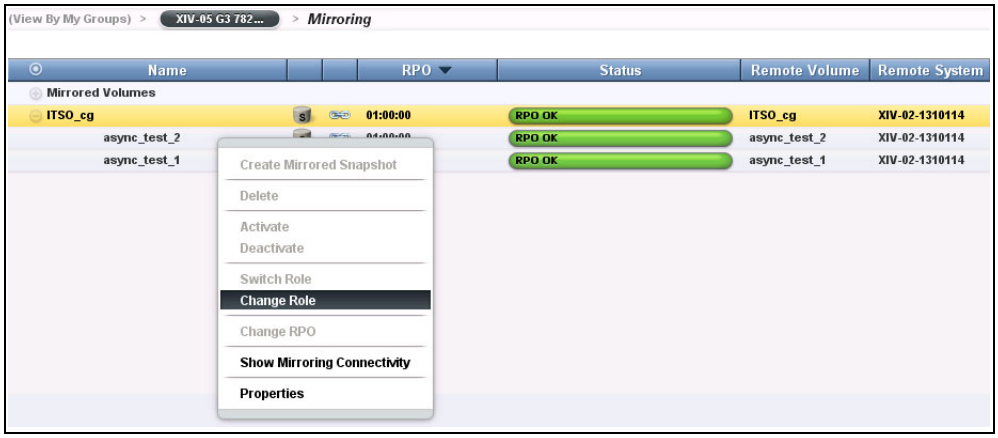


Figure 6-33 Change slave role to master

Figure 6-34 Verify change role

Figure 6-35 New master volumes

After the testing is complete the remote volumes are returned to their previous slave role (see Figure 6-36, Figure 6-37, and Figure 6-38).

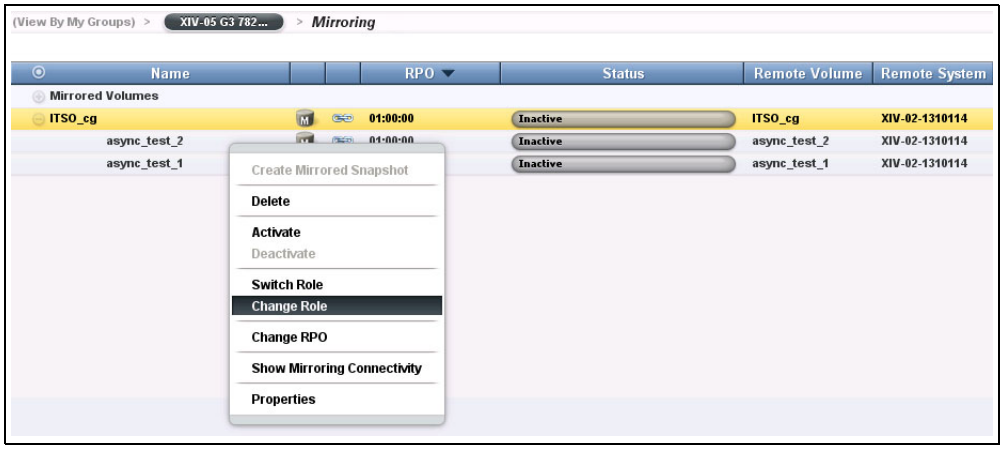


Figure 6-36 Change role back to slave

Figure 6-37 Verify change role

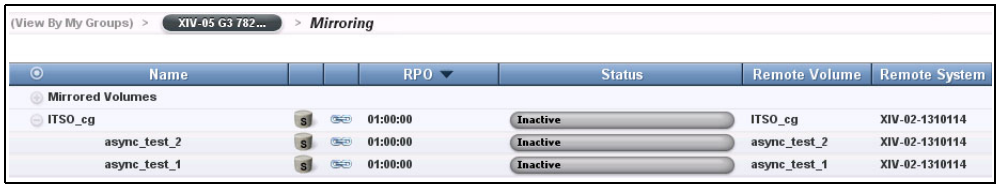


Figure 6-38 Slave role restored

Any changes made during the testing are removed by restoring the last-replicated snapshot, and new updates from the primary site will be transferred to the secondary site when the mirror is activated again, as seen in Figure 6-39 through Figure 6-41.

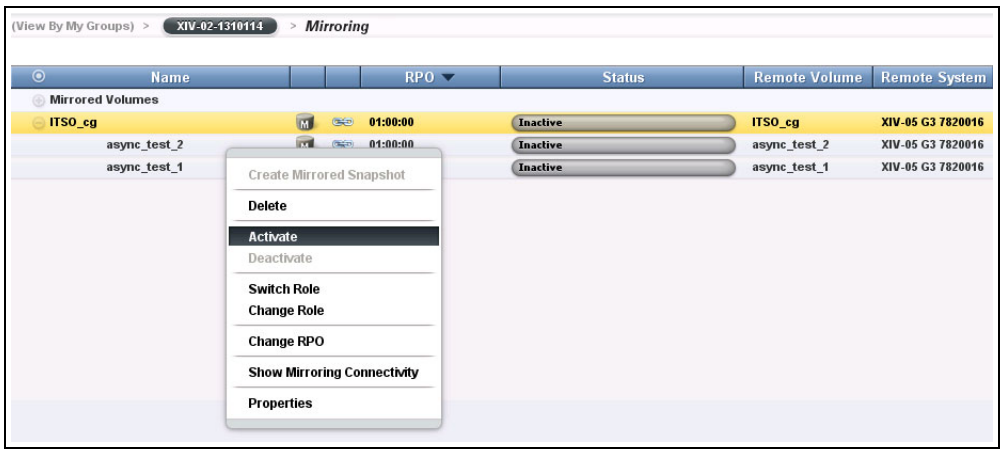


Figure 6-39 Activate mirror at primary site

(View By My Groups) > XIV-02-1310114 > Mirroring
Name
RPO
Status
Remote Volume
Remote System
Mirrored Volumes
ITSO_cg 01:00:00 RPO OK ITSO_cg XIV-05 G3 7820016
async_test_2 01:00:00 RPO OK async_test_2 XIV-05 G3 7820016
async_test_1 01:00:00 RPO OK async_test_1 XIV-05 G3 7820016

Figure 6-40 Master active

Figure 6-41 Slave active

XCLI commands for DR testing

Example 6-10 shows the steps and the corresponding XCLI commands required for DR testing.

Example 6-10 XCLI commands for DR testing

```
# Change slave to master
XIV-05 G3 7820016>>mirror_change_role cg=ITSO_cg -y
Command executed successfully.

# List mirrors with specified parameters
XIV-05 G3 7820016>>mirror_list -t local_peer_name,sync_type,current_role,target_name,active
Name                          Mirror Type        Role   Remote System   Active
ITSO_cg                       async_interval     Master XIV-02-1310114  no
async_test_1                  async_interval     Master XIV-02-1310114  no
async_test_2                  async_interval     Master XIV-02-1310114  no

# Change back to slave
XIV-05 G3 7820016>>mirror_change_role cg=ITSO_cg -y
```

```
Command executed successfully.

# List mirrors with specified parameters
XIV-05 G3 7820016>>mirror_list -t local_peer_name,sync_type,current_role,target_name,active
Name                          Mirror Type        Role   Remote System   Active
ITSO_cg                       async_interval     Slave  XIV-02-1310114  no
async_test_1                  async_interval     Slave  XIV-02-1310114  no
async_test_2                  async_interval     Slave  XIV-02-1310114  no

# Activate master on local site
XIV-02-1310114>>mirror_activate cg=ITSO_cg
Command executed successfully.

# List mirrors with specified parameters
XIV-02-1310114>>mirror_list -t local_peer_name,sync_type,current_role,target_name,active
Name                          Mirror Type        Role   Remote System       Active
ITSO_cg                       async_interval     Master XIV-05 G3 7820016   yes
async_test_1                  async_interval     Master XIV-05 G3 7820016   yes
async_test_2                  async_interval     Master XIV-05 G3 7820016   yes
```

6.8 Pool space depletion

The asynchronous mirroring process relies on special snapshots (most-recent, last-replicated) that require and consume space from the storage pool. An adequate amount of snapshot space depends on the workload characteristics and the intervals set for sync jobs. By observing applications over time, you can eventually fine-tune the percentage of pool space to reserve for snapshots.

Because the most-recent snapshot, and its subsequent promotion to be a last-replicated snapshot exists for two intervals, and that the new most-recent snapshot (prior to promotion) exists for one interval, the minimum amount of snapshot space recommended is three times the expected rate of change in the worst case interval. If the systems are approaching the border of being able to maintain RPO OK status, the snapshot allocation should be increased, because intervals might be skipped, and snapshot retention can be longer. Use prudent monitoring of snapshot utilization and appropriate management.

Tip: Set appropriate pool alert thresholds to be warned ahead of time and be able to take proactive measures to avoid any serious pool space depletion situations.

If the pool's snapshot reserve space has been consumed, replication snapshots will gradually use the remaining available space in the pool. After a single replication snapshot has been written in the regular pool's space, any new snapshot (replication snapshot or regular snapshot) will start consuming space outside the snapshot reserve. The XIV system has a sophisticated built-in multi-step process to cope with pool space depletion on the slave or on the master before it eventually deactivates the mirror. If a pool does not have enough free space to accommodate the storage requirements warranted by a new host write, the system progressively deletes snapshots within that pool until enough space is made available for successful completion of the write request. If all snapshots are deleted, the XIV Storage System might require a full re initialization of all mirrors in the group.

The process is outlined next. The system proceeds to the next step only if space continues to be insufficient to support the write-request after execution of the current step. Upon depletion of space in a pool with mirroring, the steps in Table 6-2 occur.

Table 6-2 Upon depletion of space

Step	Description
1	Deletion of unprotected[a] snapshots, first of non-mirrored volumes, then completed and outstanding Snapshot Mirrors (also known as ad-hoc sync jobs).
2	Deletion of the snapshot of any outstanding (pending) scheduled sync job.
3	Automatic deactivation of mirroring and deletion of the snapshot designated the most-recent snapshot (except for the special case described in step 5).
4	Deletion of the last-replicated snapshot.
5	Deletion of the most-recent snapshot created when activating the mirroring in Change Tracking state.
6	Deletion of protected[a] snapshots.

a. The XIV Storage System introduces the concept of *protected snapshots*. With the command pool_config_snapshots a special parameter is introduced that sets a protected priority value for snapshots in a specified pool. Pool snapshots with a delete priority value smaller than this parameter value are treated as protected snapshots and will generally be deleted only after unprotected snapshots are (with the only exception being a snapshot mirror (ad hoc) snapshot when its corresponding job is in progress). Notably, two mirroring-related snapshots will never be deleted: the last-consistent snapshot (synchronous mirroring) and the last-replicated snapshot on the slave (asynchronous mirroring).

The deletion priority of mirroring-related snapshots is set implicitly by the system and cannot be customized by the user. Consider the following information:

- The deletion priority of the asynchronous mirroring last-replicated and most-recent snapshots on the master is set to 1.
- The deletion priority of the asynchronous mirroring last-replicated snapshot on the slave and the synchronous mirroring last-consistent snapshot is set to 0.
- By default the parameter protected_snapshot_priority in pool_config_snapshots is 0.
- Non-mirrored snapshots are created by default with a deletion priority 1.

Important: If the protected_snapshot_priority in pool_config_snapshots is changed, then the system- and user-created snapshots with a deletion priority nominally equal to or lower than the protected setting will be deleted only after the internal mirroring snapshots are. This means that if the protected_snapshot_priority in pool_config_snapshots is changed to 1, then all system- and user-created snapshots with deletion priority 1 (which includes *all* snapshots created by the user if their deletion priority was not changed) will be protected and will be deleted only after internal mirroring snapshots are if pool space is depleted and the system needs to free space.

Pool space depletion on the slave

Pool space depletion on the slave means that no room is available for the last-replicated snapshot. In this case, the mirroring is deactivated.

7

Open systems considerations for Copy Services

This chapter describes the basic tasks to perform on the individual host systems when you use the XIV Copy Services.

It explains how to bring snapshot target volumes online to the same host and also to a second host. This chapter covers various UNIX platforms and VMware.

© Copyright IBM Corp. 2014. All rights reserved.

7.1 AIX specifics

This section describes the necessary steps to use volumes created by the XIV Copy Services on AIX hosts.

7.1.1 AIX and snapshots

The snapshot functionality is to copy the pointers of a source volume and create a snapshot volume. If the source volume is defined to the AIX Logical Volume Manager (LVM), all of its data structures and identifiers are copied to the snapshot also. This includes the volume group descriptor area (VGDA), which contains the physical volume identifier (PVID) and volume group identifier (VGID).

For AIX LVM, it is currently not possible to activate a volume group with a physical volume that contains a VGID and a PVID that is already used in a volume group existing on the same server. The restriction still applies even if the hdisk PVID is cleared and reassigned with the two commands listed in Example 7-1.

Example 7-1 Clearing PVIDs

```
#chdev -l <hdisk#> -a pv=clear
#chdev -l <hdisk#> -a pv=yes
```

Therefore, it is necessary to redefine the volume group information about the snapshot volumes using special procedures or the **recreatevg** command. This will alter the PVIDs and VGIDs in all the VGDAs of the snapshot volumes so that there are no conflicts with existing PVIDs and VGIDs on existing volume groups that reside on the source volumes. If you do not redefine the volume group information prior to importing the volume group, then the **importvg** command will fail.

Accessing a snapshot volume from another AIX host

The following procedure makes the data of the snapshot volume available to another AIX host that has no prior definitions of the snapshot volume in its configuration database (ODM). This host that is receiving the snapshot volumes can manage the access to these devices as described here.

If the host is using LVM or MPIO definitions that work with hdisks only, follow these steps:

1. The snapshot volume (hdisk) is newly created for AIX, and therefore the Configuration Manager should be run on the specific Fibre Channel adapter:

   ```
   #cfgmgr -l <fcs#>
   ```

2. Determine which physical volume is your snapshot volume:

   ```
   #lsdev -C |grep 2810
   ```

3. Certify that all PVIDs in all hdisks that will belong to the new volume group were set. Check this information using the **lspv** command. If they were not set, run the following command for each one to avoid failure of the **importvg** command:

   ```
   #chdev -l <hdisk#> -a pv=yes
   ```

4. Import the snapshot volume group:

   ```
   #importvg -y <volume_group_name> <hdisk#>
   ```

5. Activate the volume group (use the **varyonvg** command):

   ```
   #varyonvg <volume_group_name>
   ```

6. Verify consistency of all file systems on the snapshot volumes:

   ```
   #fsck -y <filesystem_name>
   ```

7. Mount all the snapshot file systems:

   ```
   #mount <filesystem_name>
   ```

The data is now available and you can, for example, back up the data residing on the snapshot volume to a tape device.

The disks containing the snapshot volumes might have been previously defined to an AIX system, for example, if you periodically create backups using the same set of volumes. In this case, there are two possible scenarios:

- If no volume group, file system, or logical volume structure changes were made, use “Procedure 1” on page 189 to access the snapshot volumes from the target system.
- If some modifications to the structure of the volume group were made, such as changing the file system size or modifying logical volumes (LV), use “Procedure 2” on page 189.

Procedure 1

To access the snapshot volumes from the target system if no volume group, file system, or logical volume structure changes were made, use the following steps:

1. Unmount all the source file systems:

   ```
   #umount <source_filesystem>
   ```

2. Unmount all the snapshot file systems:

   ```
   #umount <snapshot_filesystem>
   ```

3. Deactivate the snapshot volume group:

   ```
   #varyoffvg <snapshot_volume_group_name>
   ```

4. Create the snapshots on the XIV.
5. Mount all the source file systems:

   ```
   #mount <source_filesystem>
   ```

6. Activate the snapshot volume group:

   ```
   #varyonvg <snapshot_volume_group_name>
   ```

7. Perform a file system consistency check on the file systems:

   ```
   #fsck -y <snapshot_file_system_name>
   ```

8. Mount all the file systems:

   ```
   #mount <snapshot_filesystem>
   ```

Procedure 2

If some modifications have been made to the structure of the volume group, use the following steps to access the snapshot volumes:

1. Unmount all the snapshot file systems:

   ```
   #umount <snapshot_filesystem>
   ```

2. Deactivate the snapshot volume group:

   ```
   #varyoffvg <snapshot_volume_group_name>
   ```

3. Export the snapshot volume group:

   ```
   #exportvg <snapshot_volume_group_name>
   ```

4. Create the snapshots on the XIV.
5. Import the snapshot volume group:

   ```
   #importvg -y <snapshot_volume_group_name> <hdisk#>
   ```

6. Perform a file system consistency check on snapshot file systems:

   ```
   #fsck -y <snapshot_file_system_name>
   ```

7. Mount all the target file systems:

   ```
   #mount <snapshot_filesystem>
   ```

Accessing the snapshot volume from the same AIX host

This section describes a method of accessing the snapshot volume on a single AIX host while the source volume is still active on the same server. The procedure is intended to be used as a guide and might not cover all scenarios.

If you are using the same host to work with source and target volumes, you have to use the **recreatevg** command.

The **recreatevg** command overcomes the problem of duplicated LVM data structures and identifiers caused by a disk duplication process such as snapshot. It is used to recreate an AIX volume group (VG) on a set of target volumes that are copied from a set of source volumes belonging to a specific VG. The command will allocate new physical volume identifiers (PVIDs) for the member disks and a new volume group identifier (VGID) to the volume group. The command also provides options to rename the logical volumes with a prefix you specify, and options to rename labels to specify different mount points for file systems.

Accessing snapshot volumes using the recreatevg command

In this example, a volume group contains two physical volumes (hdisks), and snapshot volumes are to be created for the purpose of creating a backup.

The source volume group is `src_snap_vg`, containing hdisk2 and hdisk3.

The target volume group will be `tgt_snap_vg`; it will contain the snapshots of hdisk2 and hdisk3.

Do the following tasks to make the snapshot volumes available to AIX:

1. Stop all I/O activities and applications that access the source volumes.
2. Create the snapshot on the XIV for hdisk2 and hdisk3 with the GUI or XCLI.
3. Restart applications that access the source volumes.
4. The snapshots will now have the same volume group data structures as the source volumes hdisk2 and hdisk3. Clear the PVIDs from the target hdisks to allow a new volume group to be made:

   ```
   #chdev -l hdisk4 -a pv=clear
   #chdev -l hdisk5 -a pv=clear
   ```

5. Issue the **lspv** command; the result is shown in Example 7-2.

Example 7-2 lspv output before recreating the volume group

```
# lspv
hdisk2          00cb7f2ee8111734                    src_snap_vg     active
hdisk3          00cb7f2ee8111824                    src_snap_vg     active
hdisk4          none                                None
hdisk5          none                                None
```

6. Create the target volume group and prefix all file system path names with /backup, and prefix all AIX logical volumes with bkup:

```
recreatevg -y tgt_flash_vg -L /backup -Y bkup vpath4 vpath5
```

You must specify the hdisk names of all disk volumes participating in the volume group. The output from **lspv**, shown in Example 7-3, illustrates the new volume group definition.

Example 7-3 lspv output after recreating the volume group

```
# lspv
hdisk2          00cb7f2ee8111734                    src_snap_vg     active
hdisk3          00cb7f2ee8111824                    src_snap_vg     active
hdisk4          00cb7f2ee819f5c6                    tgt_snap_vg     active
hdisk5          00cb7f2ee819f788                    tgt_snap_vg     active
```

7. An extract from /etc/filesystems in Example 7-4 shows how **recreatevg** generates a new file system stanza. The file system named /prodfs in the source volume group is renamed to /bkp/prodfs in the target volume group. Also, the directory /bkp/prodfs is created. Notice also that the logical volume and JFS log logical volume have been renamed. The remainder of the stanza is the same as the stanza for /prodfs.

Example 7-4 Target file system stanza

```
/bkp/prodfs:
        dev             = /dev/bkupfslv01
        vfs             = jfs2
        log             = /dev/bkuploglv00
        mount           = false
        check           = false
        options         = rw
        account         = false
```

8. Perform a file system consistency check for all target file systems:

```
#fsck -y <target_file_system_name>
```

9. Mount the new file systems belonging to the target volume group to make them accessible.

7.1.2 AIX and Remote Mirroring

When you have the primary and secondary volumes in a Remote Mirror relationship, it is not possible to read the secondary volumes, unless the roles are changed from *slave* to *master*. To enable reading of the secondary volumes, they must also be synchronized. Therefore, if you are configuring the secondary volumes on the target server, it is necessary to terminate the copy pair relationship.

When the volumes are in a consistent state, the secondary volumes can be configured (**cfgmgr**) into the target system's customized device class (CuDv) of the ODM. This will bring

in the secondary volumes as hdisks, which will contain the same physical volume IDs (PVID) as the primary volumes. Because these volumes are new to the system, there is no conflict with existing PVIDs. The volume group on the secondary volumes containing the logical volume (LV) and file system information can now be imported into the Object Data Manager (ODM) and the `/etc/filesystems` file using the **`importvg`** command.

If the secondary volumes were previously defined on the target AIX system, but the original volume group was removed from the primary volumes, the old volume group and disk definitions must be removed (**`exportvg`** and **`rmdev`**) from the target volumes and redefined (**`cfgmgr`**) before running **`importvg`** again to get the new volume group definitions. If this is not done first, **`importvg`** will import the volume group improperly. The volume group data structures (PVIDs and VGID) in ODM will differ from the data structures in the VGDAs and disk volume super blocks. The file systems will not be accessible.

Making updates to the LVM information

When performing Remote Mirroring between primary and secondary volumes, the primary AIX host can create, modify, or delete existing LVM information from a volume group. However, because the secondary volume is not accessible when in a Remote Mirroring relationship, the LVM information in the secondary AIX host would be out-of-date. Therefore, scheduled periods should be allotted where write I/Os to the primary Remote Mirroring volume can be quiesced and file systems unmounted. At this point, the copy pair relationship can be terminated and the secondary AIX host can perform a *learn* on the volume group (**`importvg -L`**).

When the updates have been imported into the secondary AIX host's ODM, you can establish the Remote Mirror and Copy pair again. As soon as the Remote Mirroring pair has been established, immediately suspend the Remote Mirroring. Because there was no write I/O to the primary volumes, both the primary and secondary are consistent.

The following example shows two systems, host1 and host2, where host1 has the primary volume hdisk5 and host2 has the secondary volume hdisk16. Both systems have had their ODMs populated with the volume group itsovg from their respective Remote Mirror and Copy volumes and, prior to any modifications, both systems' ODM have the same time stamp, as shown in Example 7-5.

Example 7-5 Original time stamp

```
root@host1:/> getlvodm -T itsovg
4cc6d7ee09109a5e

root@host2:/> getlvodm -T itsovg
4cc6d7ee09109a5e
```

Volumes hdisk5 and hdisk16 are in the synchronized state, and the volume group itsovg on host1 is updated with a new logical volume. The time stamp on the VGDA of the volumes gets updated and so does the ODM on host1, but not on host2.

To update the ODM on the secondary server, it is advisable to suspend the Remote Mirror and Copy pair prior to performing the **`importvg -L`** command to avoid any conflicts from LVM actions occurring on the primary server. When the **`importvg -L`** command has completed, you can re-establish the Remote Mirror.

7.2 Copy Services using VERITAS Volume Manager

This section describes special considerations for snapshots and Remote Mirroring on Solaris systems with VERITAS Volume Manager (VxVM) support.

7.2.1 Snapshots with VERITAS Volume Manager

In many cases, a user will make a copy of a volume so that the data can be used by a different machine. In other cases, a user might want to make the copy available to the same machine. VERITAS Volume Manager assigns each disk a unique global identifier. If the volumes are on different machines, this does not present a problem. However, if they are on the same machine, you have to take some precautions. For this reason, the steps that you should take are different for the two cases.

Snapshot to a different server

One common method for making a snapshot of a VxVM volume is to freeze the I/O to the source volume, issue the snapshot, and import the new snapshot onto a second server. In general, use the following steps:

1. Unmount the target volume on Server B.
2. Freeze the I/O to the source volume on Server A.
3. Create a snapshot.
4. Thaw the I/O to the source volume on Server A.
5. Mount the target volume on Server B.

Snapshot to the same server

The simplest way to make the copy available to the source machine is to export and offline the source volumes. In Example 7-6, volume lvol is contained in Disk Group vgsnap. This Disk Group consists of two devices (xiv0_4 and xiv0_5). When those disks are taken offline, the snapshot target becomes available to the source volume, and can be imported.

Example 7-6 Making a snapshot available by exporting the source volume

```
#halt I/O on the source by unmounting the volume
umount /vol1
#create snapshot, unlock the created snapshot and map to the host here
#discover newly available disks
vxdctl enable
#deport the source volume group
vxdg deport vgsnap
#offline the source disk
vxdisk offline xiv0_4 xiv0_5
#now only the target disk is online
#import the volume again
vxdg import vgsnap
#recover the copy
vxrecover -g vgsnap -s lvol
#re-mount the volume
mount /dev/vx/dsk/vgsnap/lvol
```

If you want to make both the source and target available to the machine at the same time, changing the private region of the disk is necessary, so that VERITAS Volume Manager allows the target to be accessed as a different disk. This section explains how to simultaneously mount snapshot source and target volumes to the same host without

exporting the source volumes when using VERITAS Volume Manager. Check with VERITAS and IBM on the supportability of this method before using it.

The assumption is that the sources are constantly mounted to the Solaris host, the snapshot is performed, and the goal is to mount the copy without unmounting the source or rebooting.

Use the following procedure to mount the targets to the same host (see Example 7-7):

> **Note:** The process shown in Example 7-7 refers to the following names:
> - vgsnap2: The name of the disk group that is being created.
> - vgsnap: The name of original disk group.

1. To discover the newly available disks issue the following command:

   ```
   # vxdctl enable
   ```

2. Check that the new disks are available. The new disks are presented in output as online disks with mismatch UIDs.

   ```
   # vxdisk list
   ```

3. Import an available disk onto the host in a new disk group by using the **vxdg** command.

   ```
   # vxdg -n <name for the new disk group> -o useclonedev=on,updateid -C import
   name of the original disk group>
   ```

4. Apply the journal log to the volume located in the disk group.

   ```
   #vxrecover -g <name of new disk group> -s <name of the volume>
   ```

5. Mount the file system located in disk groups.

   ```
   # mount /dev/vx/dsk/<name of new disk group/<name of the volume> /<mount point>
   ```

Example 7-7 Importing the snapshot on same host simultaneously with using of original disk group

```
# vxdctl enable
# vxdisk list
DEVICE       TYPE            DISK         GROUP        STATUS
xiv0_0       auto:cdsdisk    vgxiv02      vgxiv        online
xiv0_4       auto:cdsdisk    vgsnap01     vgsnap       online
xiv0_5       auto:cdsdisk    vgsnap02     vgsnap       online
xiv0_8       auto:cdsdisk    -            -            online udid_mismatch
xiv0_9       auto:cdsdisk    -            -            online udid_mismatch
xiv1_0       auto:cdsdisk    vgxiv01      vgxiv        online
# vxdg -n vgsnap2 -o useclonedev=on,updateid -C import vgsnap
VxVM vxdg WARNING V-5-1-1328 Volume lvol: Temporarily renumbered due to conflict
# vxrecover -g vgsnap2 -s lvol
# mount /dev/vx/dsk/vgsnap2/lvol /test
# ls /test
VRTS_SF_HA_Solutions_5.1_Solaris_SPARC.tar
VRTSaslapm_Solaris_5.1.001.200.tar
VRTSibmxiv-5.0-SunOS-SPARC-v1_307934.tar.Z
lost+found
# vxdisk list
DEVICE       TYPE            DISK         GROUP        STATUS
xiv0_0       auto:cdsdisk    vgxiv02      vgxiv        online
xiv0_4       auto:cdsdisk    vgsnap01     vgsnap       online
xiv0_5       auto:cdsdisk    vgsnap02     vgsnap       online
xiv0_8       auto:cdsdisk    vgsnap01     vgsnap2      online clone_disk
xiv0_9       auto:cdsdisk    vgsnap02     vgsnap2      online clone_disk
xiv1_0       auto:cdsdisk    vgxiv01      vgxiv        online
```

7.2.2 Remote Mirroring with VERITAS Volume Manager

The previous section describes how to do a snapshot and mount the source and target file system on the same server. This section describes the steps necessary to mount a Remote Mirrored secondary volume onto a server that does not have sight of the primary volume. It assumes that the Remote Mirroring pair has been terminated prior to carrying out the procedure.

After the secondary volumes have been assigned, reboot the Solaris server using **reboot -- -r** or, if a reboot is not immediately possible, then issue **devfsadm**. However, reboot for reliable results.

Use the following procedure to mount the secondary volumes to another host:

1. Scan devices in the operating system device tree:

   ```
   #vxdctl enable
   ```

2. List all known disk groups on the system:

   ```
   #vxdisk -o alldgs list
   ```

3. Import the Remote Mirror disk group information:

   ```
   #vxdg -C import <disk_group_name>
   ```

4. Check the status of volumes in all disk groups:

   ```
   #vxprint -Ath
   ```

5. Bring the disk group online by using either of the following lines:

   ```
   #vxvol -g <disk_group_name> startall
   #vxrecover -g <disk_group_name> -sb
   ```

6. Perform a consistency check on the file systems in the disk group:

   ```
   #fsck -V vxfs /dev/vx/dsk/<disk_group_name>/<volume_name>
   ```

7. Mount the file system for use:

   ```
   #mount -V vxfs /dev/vx/dsk/<disk_group_name>/<volume_name> /<mount_point>
   ```

When you finish with the mirrored volume, do the following tasks:

1. Unmount the file systems in the disk group:

   ```
   #umount /<mount_point>
   ```

2. Take the volumes in the disk group offline:

   ```
   #vxvol -g <disk_group_name> stopall
   ```

3. Export disk group information from the system:

   ```
   #vxdg deport <disk_group_name>
   ```

7.3 HP-UX and Copy Services

This section describes the interaction between XIV Copy Services and Logical Volume Manager (LVM) on HP-UX. Write access to the Copy Services target volumes is either allowed for XIV Copy Services or for HP-UX. LVM commands must be used to disable host access to a volume before XIV Copy Services take control of the associated target volumes. After Copy Services have been terminated for the target volumes, LVM commands can be used to enable host access.

7.3.1 HP-UX and XIV snapshot

The following procedure must be followed to permit access to the snapshot source and target volumes simultaneously on an HP-UX host. It could be used to make an additional copy of a development database for testing or to permit concurrent development, to create a database copy for data mining that will be accessed from the same server as the OLTP data, or to create a point-in-time copy of a database for archiving to tape from the same server.

This procedure must be repeated each time you perform a snapshot and want to use the target physical volume on the same host where the snapshot source volumes are present in the Logical Volume Manager configuration. The procedure can also be used to access the target volumes on another HP-UX host.

Target preparation

Follow these steps to prepare the target system:

1. If you did not use the default Logical Volume Names (lvolnn) when they were created, create a map file of your source volume group using the **vgexport** command with the preview (**-p**) option:

   ```
   #vgexport -p -m <map file name> -p /dev/<source_vg_name>
   ```

   Tip: If the target volumes are accessed by a secondary (or target) host, this map file must be copied to the target host.

2. If the target volume group exists, remove it using the **vgexport** command. The target volumes cannot be members of a volume group when the **vgimport** command is run.

   ```
   #vgexport /dev/<target_vg_name>
   ```

3. Shut down or quiesce any applications that are accessing the snapshot source.

Snapshot execution

Follow these steps to execute and access the snapshot:

1. Quiesce or shut down the source HP-UX applications to stop any updates to the primary volumes.
2. Perform the XIV snapshot.
3. When the snapshot is finished, change the volume group ID on each DS Volume in the snapshot target. The volume ID for each volume in the snapshot target volume group must be modified on the same command line. Failure to do this will result in a mismatch of volume group IDs within the volume group. The only way to resolve this issue is to perform the snapshot again and reassign the volume group IDs using the same command line:

   ```
   vgchgid -f </dev/rdsk/c#t#d#_1>...</dev/rdsk/c#t#d#_n>
   ```

   Note: This step is not needed if another host is used to access the target devices.

4. Create the volume group for the snapshot target volumes:

   ```
   #mkdir /dev/<target_vg_name>
   #mknod /dev/<target_vg_name>/group c <lvm_major_no> <next_available_minor_no>
   ```

   Use the **lsdev -C lvm** command to determine what the major device number should be for Logical Volume Manager objects. To determine the next available minor number, examine the minor number of the group file in each volume group directory using the **ls -l** command.

5. Import the snapshot target volumes into the newly created volume group using the **vgimport** command:

   ```
   #vgimport -m <map file name> -v /dev/<target_vg_name>
   </dev/dsk/c#t#d#_1>...</dev/dsk/c#t#d#_n>
   ```

6. Activate the new volume group:

   ```
   #vgchange -a y /dev/<target_vg_name>
   ```

7. Perform a full file system check on the logical volumes in the target volume group. This is necessary to apply any changes in the JFS intent log to the file system and mark the file system as clean.

   ```
   #fsck -F vxfs -o full -y /dev/<target_vg_name>/<logical volume name>
   ```

8. If the logical volume contains a VxFS file system, mount the target logical volumes on the server:

   ```
   #mount -F vxfs /dev/<target_vg_name>/<logical volume name> <mount point>
   ```

When access to the snapshot target volumes is no longer required, unmount the file systems and deactivate (vary off) the volume group:

```
#vgchange -a n /dev/<target_vg_name>
```

If no changes are made to the source volume group prior to the subsequent snapshot, then all that is needed is to activate (vary on) the volume group and perform a full file system consistency check, as shown in steps 7 and 8.

7.3.2 HP-UX with XIV Remote Mirror

When using Remote Mirror with HP-UX, LVM handling is similar to using snapshots, apart from the fact that the volume group should be unique to the target server, so there should not be a need to perform the **vgchgid** command to change the physical volume to volume group association.

Follow these steps to bring Remote Mirror target volumes online to secondary HP-UX hosts:

1. Quiesce the source HP-UX application to cease any updates to the primary volumes.
2. Change the role of the secondary volumes to master to enable host access.
3. Rescan for hardware configuration changes using the **ioscan -fnC** disk command. Check that the disks are CLAIMED using **ioscan -funC** disk. The reason for doing this is that the volume group might have been extended to include more physical volumes.
4. Create the volume group for the Remote Mirror secondary. Use the **lsdev -C lvm** command to determine what the major device number should be for Logical Volume Manager objects. To determine the next available minor number, examine the minor number of the group file in each volume group directory using the **ls -l** command.
5. Import the Remote Mirror secondary volumes into the newly created volume group using the **vgimport** command.

6. Activate the new volume group using the **vgchange** command with the **-a y** option.
7. Perform a full file system check on the logical volumes in the target volume group. This is necessary to apply any changes in the JFS intent log to the file system and mark the file system as clean.
8. If the logical volume contains a VxFS file system, mount the target logical volumes on the server.

If changes are made to the source volume group, be sure they are reflected in the `/etc/lvmtab` of the target server. Therefore, do periodic updates to make the `lvmtab` on both source and target machines consistent. Use the previous steps, but include the following steps, before activating the volume group:

1. On the source HP-UX host, export the source volume group information into a map file using the preview option:

   ```
   #vgexport -p -m <map file name>
   ```

2. Copy the map file to the target HP-UX host.
3. On the target HP-UX host export the volume group.
4. Re-create the volume group using the HP-UX **mkdir** and **mknod** commands.
5. Import the Remote Mirror target volumes into the newly created volume group using the **vgimport** command.

When access to the Remote Mirror target volumes is no longer required, unmount the file systems and deactivate (vary off) the volume group:

```
#vgchange -a n /dev/<target_vg_name>
```

Where appropriate, reactivate the XIV Remote Mirror in normal or reverse direction. If copy direction is reversed, the master and slave roles and thus the source and target volumes are also reversed.

7.4 Windows and Copy Services

Follow these steps to mount target volumes on another Windows 2008 R2 or 2012 host:

1. Perform the Remote Mirror/Snapshot function to the target volume. Ensure that when using Remote Mirror, the target volume is in state *Consistent* for synchronous mirroring and *RPO ok* for asynchronous mirroring, and that write I/O was ceased prior to terminating the copy relationship.
2. Change the role on the target system to Master for Remote Mirroring and unlock the snapshot target volume for Snapshots if read/write access is needed.
3. Map the target volumes to the host.
4. Click **Server Manager**, click **Storage/Disk Management,** and then click **Rescan Disks**.
5. Find the disk that is associated with your volume. There are two panes for each disk; the left one says *Offline*.
6. Right-click that pane and select **Online**. The volume now has a drive letter assigned to it.

Follow these steps to mount target volumes on the same Windows 2008R2/2012 host:

1. Perform the Remote Mirror/Snapshot function onto the target volume. Ensure that when you use Remote Mirror, the target volume is in a *Consistent* state for synchronous mirroring and an *RPO OK* state for asynchronous mirroring, and that write I/O was ceased prior to terminating the copy relationship.
2. Change the role on the target system to Master for Remote Mirroring and unlock the snapshot target volume for Snapshots if read/write access is needed.
3. Map the target volumes to the host.
4. Click **Server Manager**, click **Storage/Disk Management,** and then click **Rescan Disks**.
5. Find the disk that is associated with your volume. There are two panes for each disk; the left one says *Offline*.
6. Right-click that pane and select **Online**. The volume now has another drive letter assigned to it (other than the source volume).

7.4.1 Windows Volume Shadow Copy Service with XIV Snapshot

Microsoft first introduced Volume Shadow Copy Services (VSS) in Windows 2003 Server and has included it in all subsequent releases. VSS provides a framework and the mechanisms to create consistent point-in-time copies (known as *shadow copies*) of databases and application data. It consists of a set of Microsoft COM APIs that enable volume-level snapshots to be performed while the applications that contain data on those volumes remain online and continue to write. This enables third-party software like FlashCopy Manager to centrally manage the backup and restore operation.

More details about VSS are at the following location:

`http://technet.microsoft.com/en-us/library/cc738819%28WS.10%29.aspx`

Without VSS, if you do not have an online backup solution implemented, you either must stop or quiesce applications during the backup process, or live with the side effects of an online backup with inconsistent data and open files that could not be backed up.

With VSS, you can produce consistent shadow copies by coordinating tasks with business applications, file system services, backup applications, fast recovery solutions, and storage hardware such as the XIV Storage System.

Volume Shadow Copy Services product components

Microsoft Volume Shadow Copy Services enables you to perform online backup of applications, which traditionally is not possible. VSS is supported on the XIV storage system. VSS accomplishes this by facilitating communications between the following three entities:

- Requestor

  An application that requests that a volume shadow copy be taken. These are applications, such as backup (like Tivoli Storage FlashCopy Manager) or storage management, that request a point-in-time copy of data or a shadow copy.

- Writer

  A component of an application that stores persistent information about one or more volumes that participate in shadow copy synchronization. Writers are software that is included in applications and services to help provide consistent shadow copies.

Writers serve two main purposes:

- Responding to signals provided by VSS to interface with applications to prepare for shadow copy.
- Providing information about the application name, icons, files, and a strategy to restore the files.

Writers prevent data inconsistencies. A database application (such as SQL Server or Exchange Server) or a system service (such as Active Directory) can be a writer.

- Providers

  A component that creates and maintains the shadow copies. This can occur in the software or in the hardware. For XIV, you must install and configure the IBM XIV VSS Provider.

Figure 7-1 shows the Microsoft VSS architecture and how the software provider and hardware provider interact through Volume Shadow Copy Services.

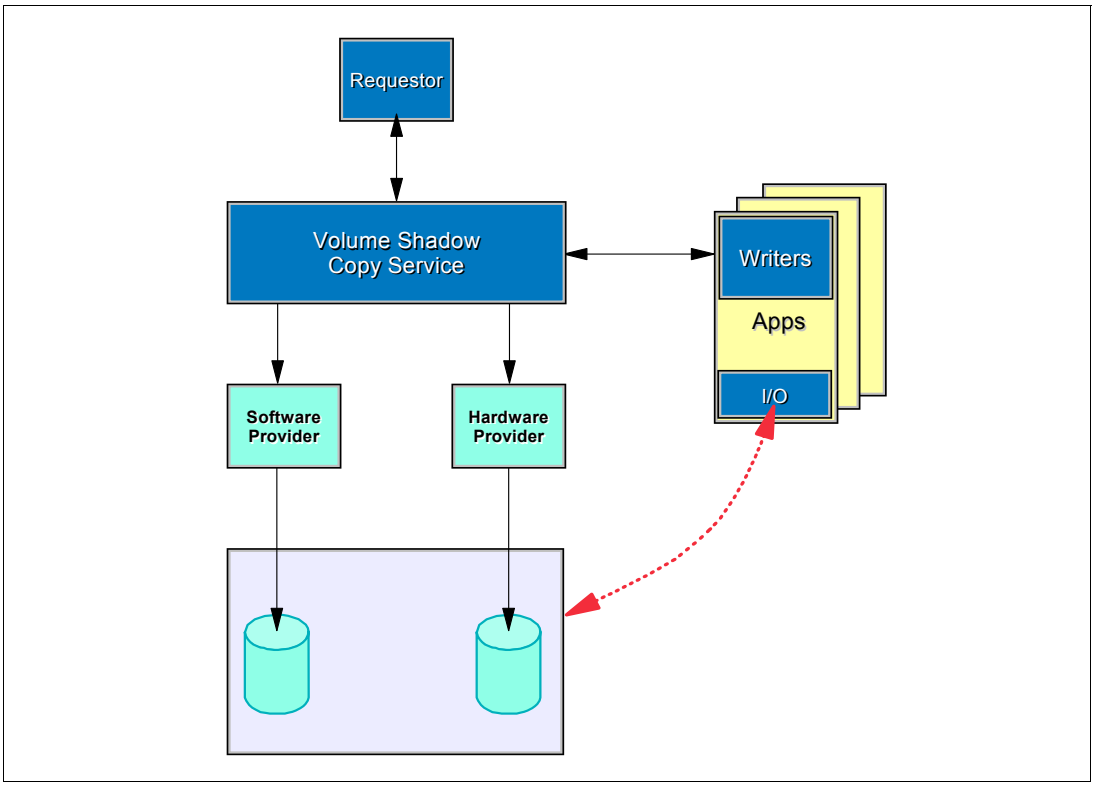


Figure 7-1 Microsoft VSS architecture

VSS uses the following terminology to characterize the nature of volumes participating in a shadow copy operation:

- Persistent

  This is a shadow copy that remains after the backup application completes its operations. This type of shadow copy also survives system reboots.

- Non-persistent

  This is a temporary shadow copy that remains only while the backup application needs it to copy the data to its backup repository.

- Transportable

  This is a shadow copy volume that is accessible from a secondary host so that the backup can be off-loaded. Transportable is a feature of hardware snapshot providers. On an XIV you can mount a snapshot volume to another host.

- Source volume

  This is the volume that contains the data to be shadow copied. These volumes contain the application data.

- Target or snapshot volume

  This is the volume that retains the shadow-copied storage files. It is an exact copy of the source volume at the time of backup.

VSS supports the following shadow copy methods:

- Clone (full copy/split mirror)

  A clone is a shadow copy volume that is a full copy of the original data as it resides on a volume. The source volume continues to take application changes while the shadow copy volume remains an exact read-only copy of the original data at the point-in-time that it was created.

- Copy-on-write (differential copy)

  A copy-on-write shadow copy volume is a differential copy (rather than a full copy) of the original data as it resides on a volume. This method makes a copy of the original data before it is overwritten with new changes. Using the modified blocks and the unchanged blocks in the original volume, a shadow copy can be logically constructed that represents the shadow copy at the point-in-time at which it was created.

- Redirect-on-write (differential copy)

  A redirect-on-write shadow copy volume is a differential copy (rather than a full copy) of the original data as it resides on a volume. This method is similar to copy-on-write, without the double-write penalty, and it offers storage-space- and performance-efficient snapshots. New writes to the original volume are redirected to another location set aside for the snapshot. The advantage of redirecting the write is that only one write takes place, whereas with copy-on-write, two writes occur (one to copy original data onto the storage space, the other to copy changed data). The XIV storage system supports redirect-on-write.

Microsoft Volume Shadow Copy Service function

Microsoft VSS accomplishes the fast backup process when a backup application initiates a shadow copy backup. Microsoft VSS coordinates with the VSS-aware writers to briefly hold writes on the databases, applications, or both. Microsoft VSS flushes the file system buffers and requests a provider to initiate a FlashCopy of the data. When the FlashCopy is logically completed, Microsoft VSS allows writes to resume and notifies the requestor that the backup has completed successfully. The volumes are mounted, hidden, and for read-only purposes, to be used when rapid restore is necessary. Alternatively, the volumes can be mounted on a different host and used for application testing or backup to tape.

The steps in the Microsoft VSS FlashCopy process are as follows:

1. The requestor notifies Microsoft VSS to prepare for shadow copy creation.
2. Microsoft VSS notifies the application-specific writer to prepare its data for making a shadow copy.
3. The writer prepares the data for that application by completing all open transactions, flushing the cache, and writing in-memory data to disk.

4. When the data is prepared for shadow copy, the writer notifies the VSS, and it relays the message to the requestor to initiate the commit copy phase.
5. VSS temporarily quiesces application I/O write requests for a few seconds and the hardware provider performs the FlashCopy on the Storage Unit.
6. After the completion of FlashCopy, VSS releases the quiesce, and database writes resume.
7. VSS queries the writers to confirm that write I/Os were successfully held during Microsoft Volume Shadow Copy.

XIV VSS Provider (xProv)

A VSS hardware provider, such as the XIV VSS Provider, is used by third-party software to act as an interface between the hardware (storage system) and the operating system. The third-party application (which can be IBM Tivoli Storage FlashCopy Manager) uses XIV VSS Provider to instruct the XIV storage system to perform a snapshot of a volume attached to the host system.

XIV VSS provider installation

This section illustrates the installation of the XIV VSS Provider. At the time of writing, the XIV VSS Provider 2.4.0 version was available. Version 2.3.2 added support for Windows 2012.We used a Windows 2008 R2 SP1 64-bit host and a VSS Provider 2.3.1 for our tests. To obtain the system requirements, see the *IBM VSS Provider - Xprov Release Notes*, which includes a chapter about the system requirements. Download the XIV VSS Provider version and release notes from the following location:

`http://ibm.co/1fmOIMs`

The installation of the XIV VSS Provider is a straightforward Windows application installation:

1. Locate the XIV VSS Provider installation file, also known as the *xProv* installation file. If the XIV VSS Provider 2.3.1 is downloaded from the Internet, the file name is `xProvSetup-2.3.1-x64.exe`. Execute the file to start the installation.
2. A Welcome window opens as shown in Figure 7-2 on page 203. Click **Next**.

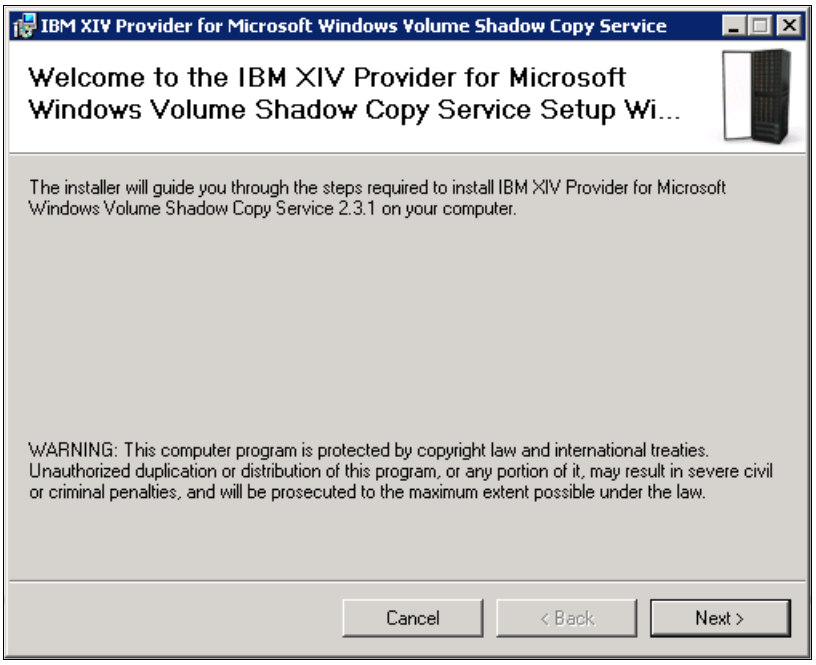

IBM XIV Provider for Microsoft Windows Volume Shadow Copy Service
Welcome to the IBM XIV Provider for Microsoft Windows Volume Shadow Copy Service Setup Wi...
The installer will guide you through the steps required to install IBM XIV Provider for Microsoft Windows Volume Shadow Copy Service 2.3.1 on your computer.
WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.
Cancel
< Back
Next >

Figure 7-2 XIV VSS provider installation: Welcome window

3. The License Agreement window is displayed. To continue the installation you must accept the license agreement.
4. Specify the XIV VSS Provider configuration file directory and the installation directory. Keep the default directory folder and installation folder or change it to meet your needs.
5. A dialog window for post-installation operations is opened, as shown in Figure 7-3. You can perform a post-installation configuration during the installation process or at a later time. When done, click **Next**.

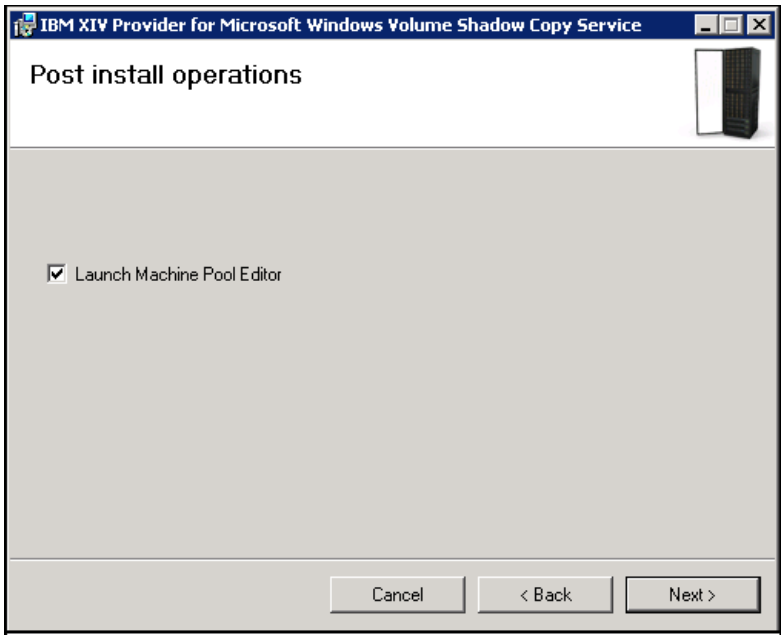

IBM XIV Provider for Microsoft Windows Volume Shadow Copy Service
Post install operations
Launch Machine Pool Editor
Cancel
< Back
Next >

Figure 7-3 Installation: Post-installation operation

6. A Confirm Installation window is displayed. You can go back to make changes if required, or confirm the installation by clicking **Next**.
7. Click **Close** to exit after the installation is complete.

XIV VSS Provider configuration

Configure the XIV VSS Provider using the following steps:

1. If the post installation check box was selected during installation (Figure 7-3 on page 203), the XIV VSS Provider configuration window is shown.
2. Right-click the Machine Pool Editor.
3. In the dialog shown in Figure 7-4, click **New System** to open the New System window.

Figure 7-4 XIV Configuration: Machine Pool Editor

4. The Add System Management window shown in Figure 7-5 is displayed. Enter the user name and password of an XIV user with administrator privileges (storageadmin role) and the primary IP address of the XIV Storage System. If the snapshot is taken of a volume that is in a mirror relation and you want to have the snapshot on source and target systems, then select **Enable Replicated Snapshots** and click **Add**.

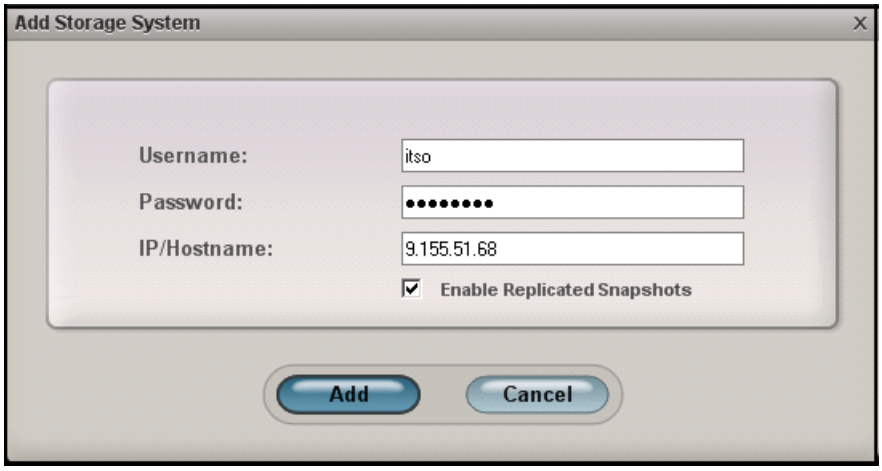


Figure 7-5 XIV configuration: Add machine

5. You are returned to the VSS Machine Pool Editor window. The VSS Provider collected additional information about the XIV storage system, as illustrated in Figure 7-6.

Figure 7-6 XIV Configuration: Machine Pool Editor

6. At this point the XIV VSS Provider configuration is complete and you can close the Machine Pool Editor window. If you must add other XIV Storage Systems, repeat steps 3 - 5.

After the XIV VSS Provider has been configured, ensure that the operating system can recognize it. To do this, launch the **vssadmin list providers** command from the operating system command line.

Make sure that `IBM XIV VSS HW Provider` is in the list of installed VSS providers returned by the **vssadmin** command as shown in Example 7-8.

Example 7-8 Output of vssadmin command

```
c:\Users\Administrator>vssadmin list providers
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2005 Microsoft Corp.

Provider name: 'Microsoft Software Shadow Copy provider 1.0'
   Provider type: System
   Provider Id: {b5946137-7b9f-4925-af80-51abd60b20d5}
   Version: 1.0.0.7

Provider name: 'IBM XIV VSS HW Provider'
   Provider type: Hardware
   Provider Id: {d51fe294-36c3-4ead-b837-1a6783844b1d}
   Version: 2.3.1
```

Diskshadow command line utility

All editions of Windows Server 2008 contain a command line utility (`DiskShadow.exe`) for the creation, deletion, and restoration of shadow copies (snapshots). It is the first integrated VSS requestor that can create hardware shadow copies and one of many utilities for validating VSS operations. The tool is similar to **vshadow** (a tool included with the Volume Shadow Copy/VSS SDK), but has an interface similar to **diskpart** utility. More details about **diskshadow** are at the following location:

http://technet.microsoft.com/en-us/library/cc772172%28WS.10%29.aspx

The steps to test the creation of a persistent snapshot of a basic disk on XIV vis VSS are shown in Example 7-9. The snapshot will be automatically unlocked and mapped to the server. Assign a drive letter to the volume and access the data on the file system.

Example 7-9 Diskshadow snapshot creation

```
c:\Users\Administrator>diskshadow
Microsoft DiskShadow version 1.0
Copyright (C) 2007 Microsoft Corporation
On computer:  WIN-9E9FNBTKC48,  9/9/2011 2:47:04 PM
DISKSHADOW> set context persistent
DISKSHADOW> add volume z:
DISKSHADOW> create
Alias VSS_SHADOW_1 for shadow ID {6efc9c8d-ebff-4bf5-86d9-2bb35d4204b0} set as e
nvironment variable.
Alias VSS_SHADOW_SET for shadow set ID {a9ec8dcb-f7b8-4819-9560-917ffcdffd18} se
t as environment variable.
Querying all shadow copies with the shadow copy set ID {a9ec8dcb-f7b8-4819-9560-
917ffcdffd18}
        * Shadow copy ID = {6efc9c8d-ebff-4bf5-86d9-2bb35d4204b0}
```

```
%VSS_SHADOW_1%
                - Shadow copy set: {a9ec8dcb-f7b8-4819-9560-917ffcdffd18}
%VSS_SHADOW_SET%
                - Original count of shadow copies = 1
                - Original volume name: \\?\Volume{f17d278a-0b38-42c3-bdf1-1a0f9
1fb6686}\ [Z:\]
                - Creation time: 9/9/2011 2:48:30 PM
                - Shadow copy device name: \\?\Volume{8b4bb63a-d942-11e0-b2e6-02
215e92a303}
                - Originating machine: WIN-9E9FNBTKC48
                - Service machine: WIN-9E9FNBTKC48
                - Not exposed
                - Provider ID: {d51fe294-36c3-4ead-b837-1a6783844b1d}
                - Attributes:  No_Auto_Release Persistent Hardware
Number of shadow copies listed: 1
```

The snapshot with this Shadow Copy ID is visible as depicted in Figure 7-7.

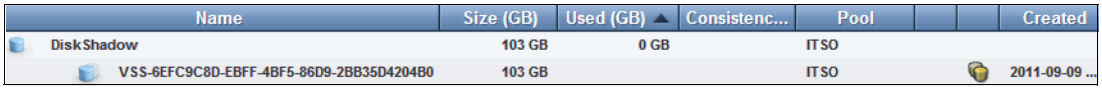


Figure 7-7 VSS snapshot

Mirrored VSS snapshot creation

Starting with XIV VSS Provider version 2.2.4 it is possible to create snapshots through VSS on a mirrored XIV volume. Before you start the VSS snapshot creation the mirror relation must exist and be active. The VSS process will create a snapshot of the source volume and also the target volume. Use the following steps to create the snapshot on both mirror sites:

1. Add the target XIV system also to the XIV VSS Provider and select **Enable Replicated Snapshots.** If the system was added without this option, remove it at first.
2. Set up XIV Volume mirroring for the volumes you want to use for VSS snapshot replication.
3. Run a VSS create operation on the mirrored LUN.

To be able to import the shadow copy to the same or a different computer the `transportable` option must be used and the `example1.cab` is needed, as shown in Example 7-10. The import works on basic disks only.

Example 7-10 VSS snapshot creation of a mirrored volume

```
c:\Users\Administrator>diskshadow
Microsoft DiskShadow version 1.0
Copyright (C) 2007 Microsoft Corporation
On computer:  WIN-9E9FNBTKC48,  9/12/2011 1:17:47 PM
DISKSHADOW> set context persistent
DISKSHADOW> set option transportable
DISKSHADOW> set metadata c:\Users\Administrator\example1.cab
DISKSHADOW> add volume z:
DISKSHADOW> create
Alias VSS_SHADOW_1 for shadow ID {07a8d8f3-89a1-421f-8dac-e6c2821e1a88} set as environment variable.
Alias VSS_SHADOW_SET for shadow set ID {ae2124f8-467f-4b05-9bde-4cad40a26130} set as environment
variable.
```

The snapshots created by VSS on the source and target XIV storage system are depicted in Figure 7-8 and Figure 7-9.

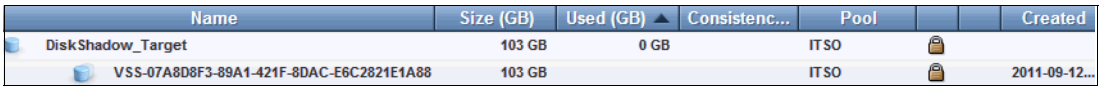


Figure 7-8 VSS mirrored snapshot source

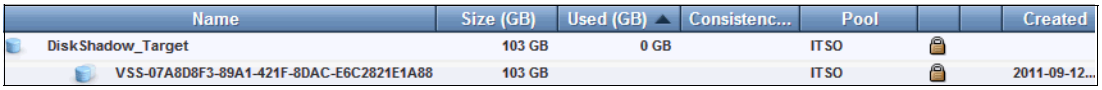


Figure 7-9 VSS mirrored snapshot target

To test the import of the data afterwards to another server, copy the `example1.cab` file to this server. The host and its ports must be defined on the XIV storage system. The commands to load the metadata and import the VSS snapshot to the server are shown in Example 7-11. Afterwards assign a drive letter to the volume and access the data on the file system.

Example 7-11 diskshadow import

```
C:\Users\Administrator>diskshadow
Microsoft DiskShadow version 1.0
Copyright (C) 2007 Microsoft Corporation
On computer:  WIN-6DQF6JFOQH7,  9/12/2011 1:43:47 PM
DISKSHADOW> load metadata example1.cab
Alias VSS_SHADOW_1 for value {07a8d8f3-89a1-421f-8dac-e6c2821e1a88} set as an environment variable.
Alias VSS_SHADOW_SET for value {ae2124f8-467f-4b05-9bde-4cad40a26130} set as an environment
variable.
DISKSHADOW> import
```

7.5 VMware virtual infrastructure and Copy Services

The section is not intended to cover every possible use of Copy Services with VMware; rather, it is intended to provide hints and tips that will be useful in many different Copy Services scenarios.

When using Copy Services with the guest operating systems, the restrictions of the guest operating system still apply. In some cases, using Copy Services in a VMware environment might impose additional restrictions.

7.5.1 Virtual machine considerations regarding Copy Services

Before creating snapshot, it is important to prepare both the source and target machines to be copied. For the source machine, this typically means quiescing the applications, unmounting the source volumes, and flushing memory buffers to disk. See the appropriate sections for your operating systems for more information about this topic.

For the target machine, typically the target volumes must be unmounted. This prevents the operating system from accidentally corrupting the target volumes with buffered writes, and also preventing users from accessing the target LUNs until the snapshot is logically complete.

With VMware, there is an additional restriction that the target virtual machine must be shut down before issuing the snapshot. VMware also performs caching, in addition to any caching

the guest operating system might do. To be able to use the FlashCopy target volumes with ESX Server, you need to ensure that the ESX Server can see the target volumes. In addition to checking the SAN zoning and the host attachment within the XIV, you might need a SAN rescan issued by the Virtual Center.

If the snapshot LUNs contain a VMFS file system, the ESX host will detect this on the target LUNs and add them as a new data store to its inventory. The VMs stored on this data store can then be opened on the ESX host. To assign the existing virtual disks to new VMs, in the Add Hardware Wizard panel, select **Use an existing virtual disk** and choose the `.vmdk` file you want to use. See Figure 7-10.

If the snapshot LUNs were assigned as RDMs, the target LUNs can be assigned to a VM by creating a new RDM for this VM. In the Add Hardware Wizard panel, select **Raw Device Mapping** and use the same parameters as on the source VM.

Note: If you do not shut down the source VM, reservations might prevent you from using the target LUNs.

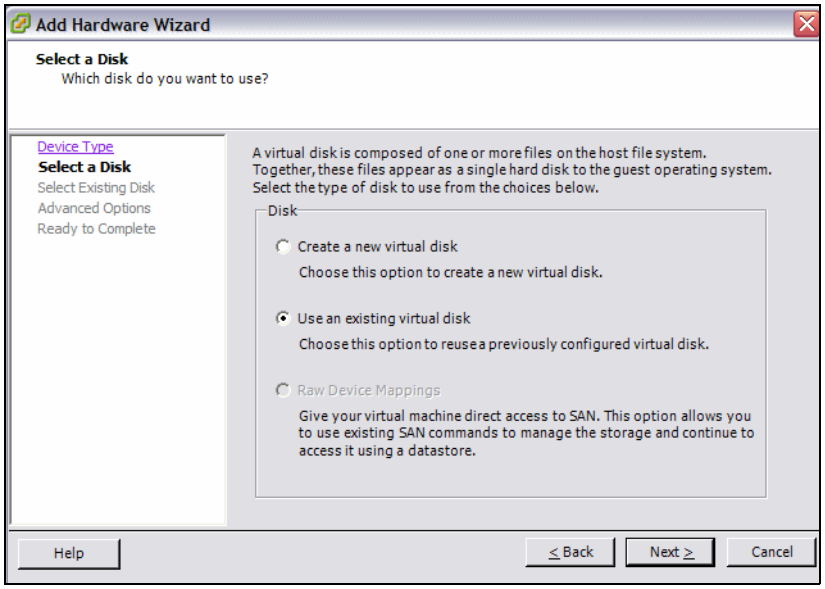

Add Hardware Wizard
Select a Disk
Which disk do you want to use?
Device Type
Select a Disk
Select Existing Disk
Advanced Options
Ready to Complete
A virtual disk is composed of one or more files on the host file system.
Together, these files appear as a single hard disk to the guest operating system.
Select the type of disk to use from the choices below.
Disk
Create a new virtual disk
Choose this option to create a new virtual disk.
Use an existing virtual disk
Choose this option to reuse a previously configured virtual disk.
Raw Device Mappings
Give your virtual machine direct access to SAN. This option allows you to use existing SAN commands to manage the storage and continue to access it using a datastore.
Help
< Back
Next >
Cancel

Figure 7-10 Adding an existing virtual disk to a VM

7.5.2 VMware ESX server and snapshots

In general, snapshots can be used within VMware virtual infrastructure in the following ways:

- On raw LUNs that are attached through RDM to a host
- On LUNs that are used to build up VMFS data stores which store VMs and virtual disks

Snapshots on LUNs used for VMFS data stores

Since version 3, all files a virtual machine is made up from are stored on VMFS partitions (usually that is configuration, BIOS, and one or more virtual disks). Therefore, the whole VM is most commonly stored in one single location. Because snapshot operations are always done on a whole volume, this provides an easy way to create point-in-time backups of whole virtual machines. Nevertheless you have to make sure that the data on the VMFS volume is consistent, and thus the VMs located on this data store must be shut down before initiating the snapshot on XIV.

Because a VMFS data store can contain more than one LUN, the user has to make sure all participating LUNs are mirrored using snapshot to get a complete copy of the data store.

Figure 7-11 shows an ESX host with 2 virtual machines, each using one virtual disk. The ESX host has one VMFS data store consisting of 2 XIV LUNs “1” and “2.” To get a complete copy of the VMFS data store, both LUNs must be placed into consistency group and then a snapshot is taken. Using snapshots on VMFS LUNs, it is easy to create backups of whole VMs.

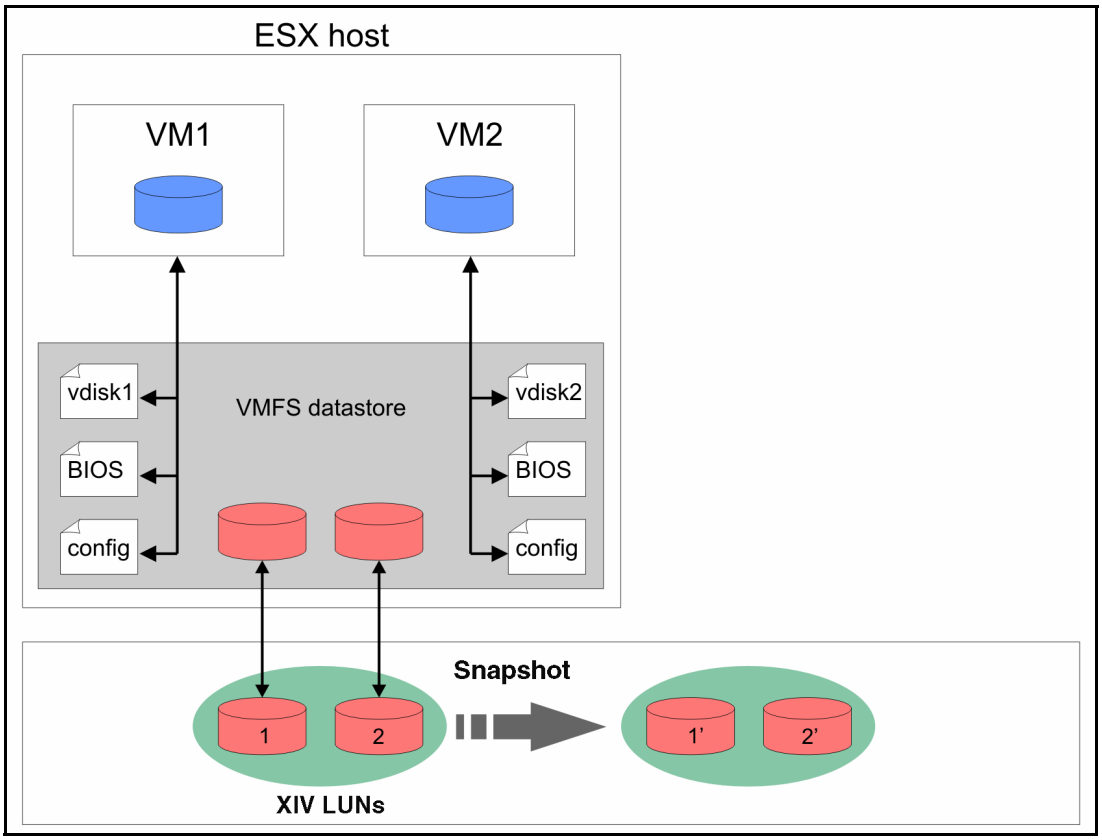


Figure 7-11 Using snapshot on VMFS volumes

Snapshot on LUNs used for RDM

Raw device mappings (RDM) can be done in two ways:

- In *physical mode*, the LUN is mostly treated like any other physical LUN. In virtual mode the virtualization layer provides features like snapshots that are normally only available for virtual disks.
- In *virtual compatibility mode* you have to make sure that the LUN you are going to copy is in a consistent state. Depending on the disk mode and current usage you might have to append the redo-log first to get a usable copy of the disk. If *persistent* or *non persistent* mode is used, the LUN can be handled like an RDM in physical compatibility mode.

For details and restrictions, see the VMware *Fibre Channel SAN Configuration Guide*:

http://pubs.vmware.com/vsp40u1_e/wwhelp/wwhimpl/js/html/wwhelp.htm#href=fc_san_config/esx_san_config_fc.1.1.html

The following paragraphs are valid for both compatibility modes. However, keep in mind that extra work on the ESX host or VMs might be required for the virtual compatibility mode.

Using snapshot within a virtual machine

In Figure 7-12, a LUN that is assigned to a VM through RDM is copied using snapshot on an IBM XIV Storage System. The target LUN is then assigned to the same VM by creating a second RDM. After issuing the snapshot job HDD1 and HDD2 have the same content.

For virtual disks, this can be achieved simply by copying the `.vmdk` files on the VMFS data store. However, the copy is not available instantly as with snapshot; instead, you will have to wait until the copy job has finished duplicating the whole `.vmdk` file.

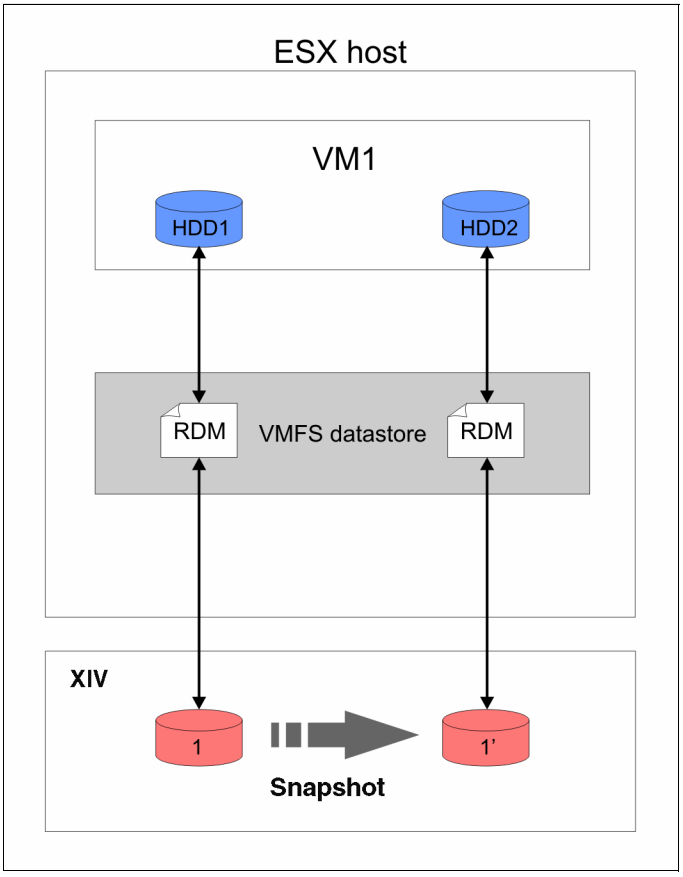


Figure 7-12 Using snapshot within a VM: HDD1 is the source for target HDD2

Using snapshot between two virtual machines

This works in the same way as using snapshot within a virtual machine, but the target disks are assigned to another VM this time. This might be useful to create clones of a VM.

After issuing the snapshot job, LUN 1’ can be assigned to a second VM, which then can work with a copy of VM1's HDD1 (Figure 7-13).

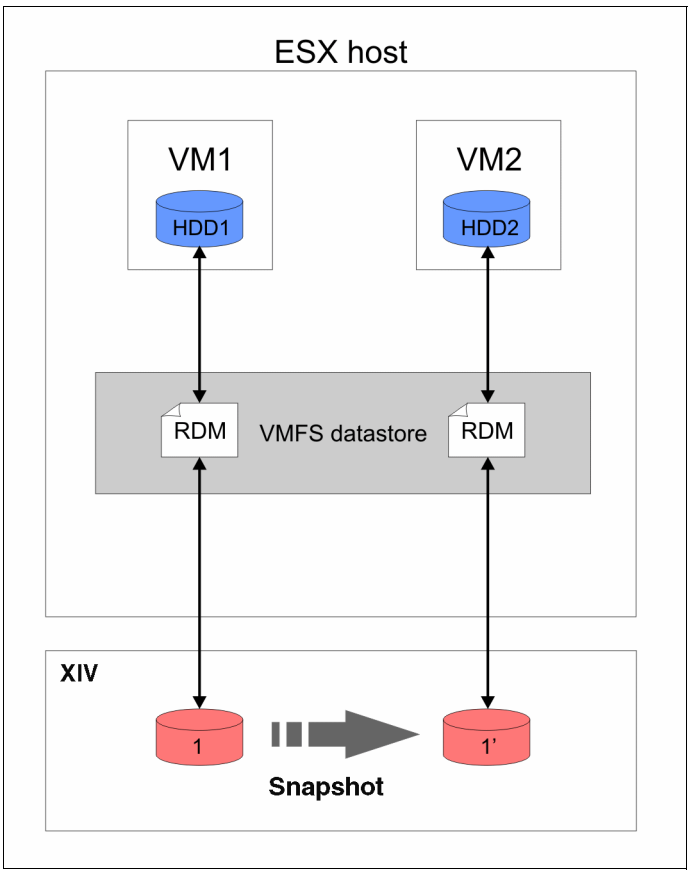


Figure 7-13 Using snapshot between two different VMs: VM1's HDD1 is the source for HDD2 in VM2

Using snapshot between ESX Server hosts

This scenario shows how to use the target LUNs on a different ESX Server host. This is especially useful for disaster recovery if one ESX Server host fails for any reason. If LUNs with VMFS are duplicated using snapshot, it is possible to create a copy of the whole virtual environment of one ESX Server host that can be migrated to another physical host with only minimal effort.

To be able to do this, both ESX Server hosts must be able to access the same snapshot LUN (Figure 7-14).

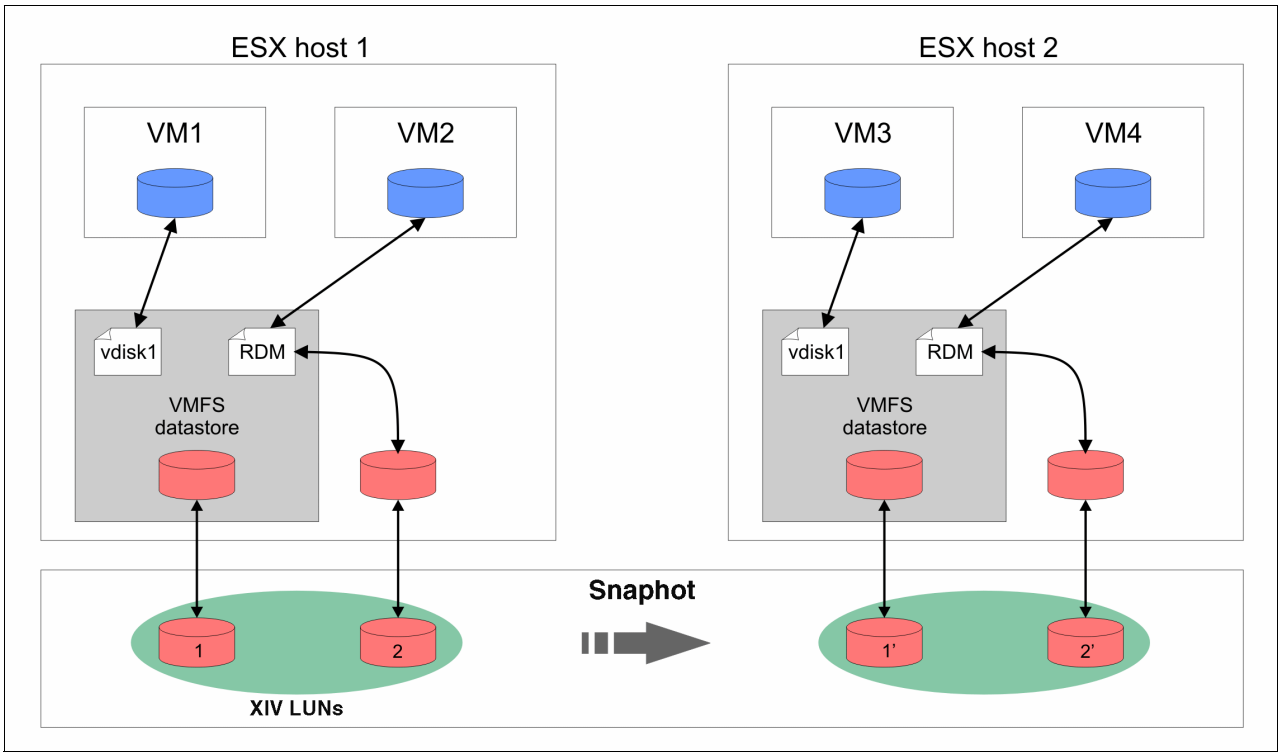


Figure 7-14 Snapshot between 2 ESX hosts

Figure 7-14 shows using snapshot on a consistency group that includes two volumes. LUN 1 is used for a VMFS data store whereas LUN 2 is assigned to VM2 as an RDM. These two LUNs are then copied with snapshot and attached to another ESX Server host. In ESX host 2 we now assign the VDisk that is stored on the VMFS partition on LUN 1' to VM3 and attach LUN 2 through RDM to VM4. By doing this we create a copy of ESX host 1's virtual environment and use it on ESX host 2.

Note: If you use snapshot on VMFS volumes and assign them to the same ESX Server host, the server does not allow the target to be used because the VMFS volume identifiers have been duplicated. To circumvent this, VMware ESX server provides the possibility of *VMFS Volume Resignaturing*. For details about resignaturing, see the "Managing Duplicate VMFS Datastores" topic in the *Fibre Channel SAN Configuration Guide*:

`http://pubs.vmware.com/vsp40u1_e/wwhelp/wwhimpl/js/html/wwhelp.htm#href=fc_san_config/c_managing_duplicate_vmfs_datastores.html`

Using VSS snapshots on Windows VMs to do online backups of applications

VSS provides a framework and the mechanisms to create consistent point-in-time copies (known as *shadow copies*) of databases and application data without the need to shut down the application or the VM. More details about VSS are in 7.4.1, "Windows Volume Shadow Copy Service with XIV Snapshot" on page 199.

At the time of writing, the XIV VSS Provider 2.4.0 version was available. Since version 2.3.0, you can do VSS snapshots of raw device mappings in Windows VMs. Version 2.3.2 added support for vSphere 5.0 and 5.1 platforms. We used XIV VSS Provider 2.3.1 on Windows 2008 R2 SP1 64-bit VM for our tests.

The XIV VSS Hardware Provider version, release notes, and installation guide can be downloaded from the following location:

http://ibm.co/1fmOIMs

Use the following steps to create a VSS snapshot of a basic disk on a Windows VM with the XIV VSS provider. Steps 1 - 4, although included here, are normally done before installing the VSS provider.

1. Create a host on XIV, add the ports to it, and map a LUN to the ESX or ESXi server.
2. Do a raw device mapping of the LUN in *physical mode* to the Windows VM.
3. Do a rescan on the Windows VM, if necessary.
4. Bring the disk online, initialize it and create a file system on it.
5. Install the XIV VSS provider and configure the XIV Storage System as described in “XIV VSS Provider (xProv)” on page 202.
6. Enter credentials for the ESX or vCenter server by MachinePoolCLI as shown in Example 7-12. The credentials are needed to perform the following tasks on the ESX or vCenter server:
   a. Select **Host** → **Configuration** → **Storage partition configuration**.
   b. Select **Virtual machine** → **Configuration** → **Raw device**.
   c. Select **Virtual machine** → **Configuration** → **Change resource**.
   d. Select **Virtual machine** → **Configuration** → **Add or remove device**.

Example 7-12 Adding an ESX or vCenter server to XIV VSS provider

```
C:\Program Files\IBM\IBM XIV Provider for Microsoft Windows Volume Shadow Copy S
ervice\.NET>MachinePoolCLI.exe /ae Administrator Test1234a 9.155.113.142
Connecting to ESX Server or vCenter with SSL......
Successfully connected to ESX Server or vCenter.
Successfully added ESX Server or vCenter, url:'https://9.155.113.142/sdk', user:
 'Administrator'
```

7. Create an application-consistent snapshot through your VSS requestor (such as Tivoli FlashCopy Manager). The steps to test the creation of a persistent snapshot of a basic disk which is mapped as raw device by ESX server are shown in Example 7-13. The snapshot will be automatically unlocked and mapped to the server. Furthermore, the ESX server will do a rescan and map the snapshot to the Windows VM. Assign a drive letter to the volume and access the data on the file system.

Example 7-13 Creation of a VSS snapshot on a Windows VM

```
C:\Users\Administrator>diskshadow
Microsoft DiskShadow version 1.0
Copyright (C) 2007 Microsoft Corporation
On computer:  WIN-GJ5E8KR49EE,  9/16/2011 5:53:57 PM
DISKSHADOW> set context persistent
DISKSHADOW> add volume e:
DISKSHADOW> create
Alias VSS_SHADOW_1 for shadow ID {34b30cbc-79c4-4b3b-b906-671cd0ba84fa} set as
environment variable.
```

```
Alias VSS_SHADOW_SET for shadow set ID {26e7cd2c-e0a8-4df5-acf0-d12ee06b9622} set as
environment variable.
Querying all shadow copies with the shadow copy set ID {26e7cd2c-e0a8-4df5-acf0-
d12ee06b9622}
        * Shadow copy ID = {34b30cbc-79c4-4b3b-b906-671cd0ba84fa}
%VSS_SHADOW_1%
                - Shadow copy set: {26e7cd2c-e0a8-4df5-acf0-d12ee06b9622}
%VSS_SHADOW_SET%
                - Original count of shadow copies = 1
                - Original volume name: \\?\Volume{8dd4b9f2-e076-11e0-a391-00505
6a6319f}\ [E:\]
                - Creation time: 9/16/2011 5:55:00 PM
                - Shadow copy device name: \\?\Volume{8dd4ba27-e076-11e0-a391-00
5056a6319f}
                - Originating machine: WIN-GJ5E8KR49EE
                - Service machine: WIN-GJ5E8KR49EE
                - Not exposed
                - Provider ID: {d51fe294-36c3-4ead-b837-1a6783844b1d}
                - Attributes:  No_Auto_Release Persistent Hardware
Number of shadow copies listed: 1
```

The snapshot with this Shadow Copy ID is visible as depicted in Figure 7-7 on page 206.

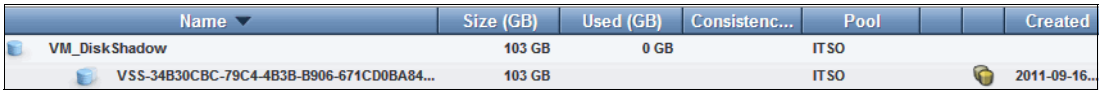


Figure 7-15 VSS snapshot of a Windows VM raw device mapping

7.5.3 ESX and Remote Mirroring

It is possible to use Remote Mirror with all three types of disks. However, in most environments, raw System LUNs in physical compatibility mode are preferred.

As with snapshots, using VMware with Remote Mirror contains all the advantages and limitations of the guest operating system. See the individual guest operating system sections for relevant information.

However, it might be possible to use raw System LUNs in physical compatibility mode. Check with IBM on the supportability of this procedure.

At a high level, the steps for creating a Remote Mirror are as follows:

1. Shut down the guest operating system on the target ESX Server.
2. Establish remote mirroring from the source volumes to the target volumes.
3. When the initial copy has completed and the volumes are synchronized, suspend or remove the Remote Mirroring relationships.
4. Issue the **Rescan** command on the target ESX Server.
5. Assign the mirrored volumes to the target virtual machine if they are not already assigned to the it. Virtual disks on VMFS volumes should be assigned as existing volumes, whereas raw volumes should be assigned as RDMs using the same parameters as on the source host.
6. Start the virtual machine and, if necessary, mount the target volumes.

Figure 7-16 shows a scenario similar to the one in Figure 7-14 on page 212, but now the source and target volumes are located on two separate XIVs. This setup can be used for disaster recovery solutions where ESX host 2 would be located in the backup data center.

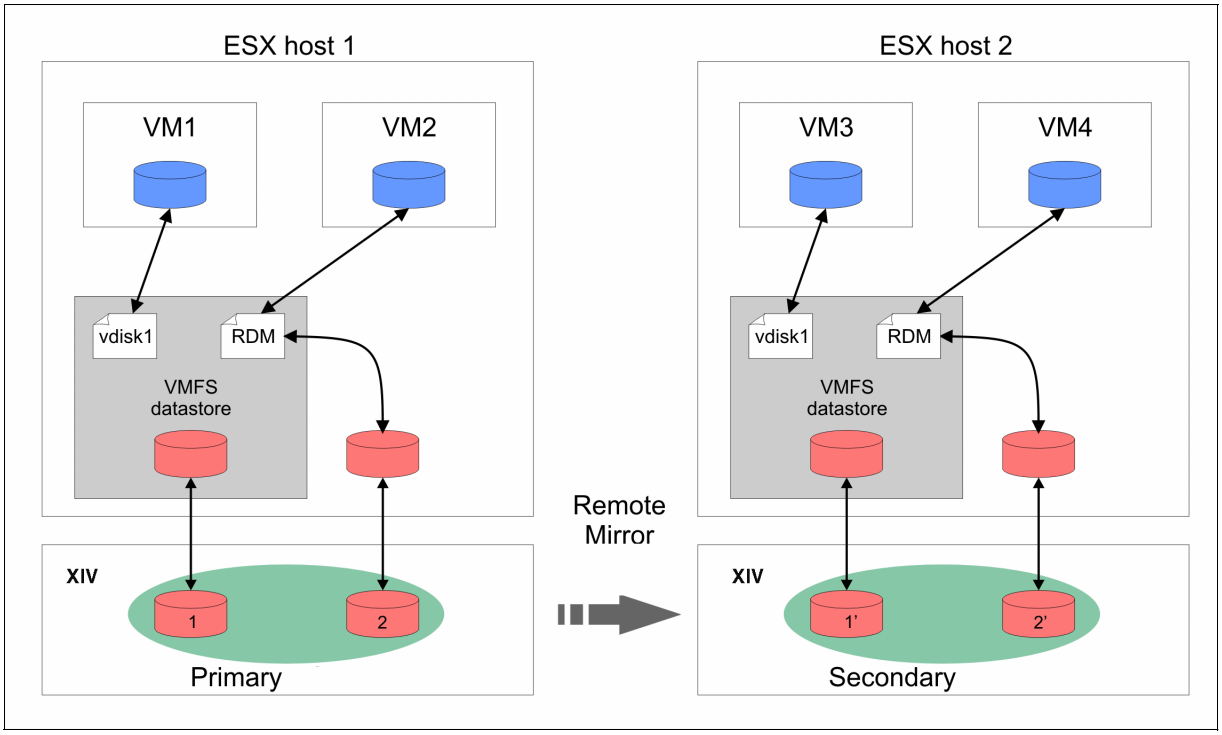


Figure 7-16 Using Remote Mirror and Copy functions

In addition, integration of VMware Site Recovery Manager with IBM XIV Storage System over IBM XIV Site Replication Adapter (SRA) for VMware SRM is supported. For more details about XIV SRA and VMware SRM see *XIV Storage System in a VMware Environment*, REDP-4965.

8

IBM i considerations for Copy Services

This chapter describes the basic tasks to do on IBM i systems when you use the XIV Copy Services.

Several illustrations in this chapter are based on a previous version of the XIV GUI.

© Copyright IBM Corp. 2014. All rights reserved.

8.1 IBM i functions and XIV as external storage

To better understand solutions using IBM i and XIV, it is necessary to have basic knowledge of IBM i functions and features that enable external storage implementation and use.

The following functions are discussed in this section:

- IBM i structure
- Single level storage

8.1.1 IBM i structure

IBM i is the newest generation of operating system previously known as IBM AS/400® or I5/OS. It runs in a partition of IBM POWER® servers or Blade servers, and also on IBM System i® and some IBM System p models.

A partition of POWER server can host one of the following operating systems that is configured and managed through a Hardware Management Console (HMC) that is connected to the IBM i through an Ethernet connection:

- IBM i
- Linux
- AIX

The remainder of this chapter refers to an IBM i partition in a POWER server or blade server simply as a *partition*.

8.1.2 Single-level storage

IBM i uses *single-level storage* architecture. This means that the IBM i sees all disk space and the main memory as one storage area, and uses the same set of virtual addresses to cover both main memory and disk space.

Paging in this virtual address space is performed in 4 KB pages. Single-level storage is graphically depicted in Figure 8-1.

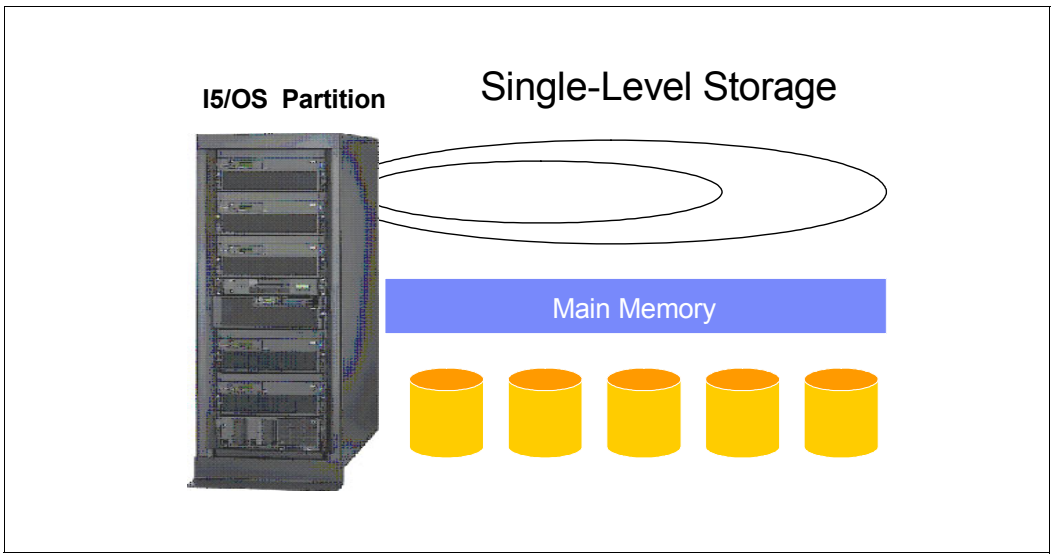


Figure 8-1 Single-level storage

When the application performs an input/output (I/O) operation, the portion of the program that contains read or write instructions is first brought into main memory, where the instructions are then executed.

With the read request, the virtual addresses of the needed record are resolved and for each page needed, storage management first checks whether it is in the main memory. If the page is there, it is used for resolving the read request. But if the corresponding page is not in the main memory, it must be retrieved from disk (page fault). When a page is retrieved, it replaces a page that was not recently used; the replaced page is swapped to disk.

Similarly, writing a new record or updating an existing record is done in main memory, and the affected pages are marked as changed. A changed page remains in main memory until it is swapped to disk as a result of a page fault. Pages are also written to disk when a file is closed or when write to disk is forced by a user through commands and parameters. Also, database journals are written to the disk.

An *object* in IBM i is anything that exists and occupies space in storage and on which operations can be performed. For example, a library, a database file, a user profile, a program are all objects in IBM i.

8.1.3 Auxiliary storage pools (ASPs)

IBM i has a rich storage management heritage. From the start, the System i platform made managing storage simple through the use of disk pools. For most customers, this meant a single pool of disks called the System Auxiliary Storage Pools (ASPs). Automatic use of newly added disk units, RAID protection, and automatic data spreading, load balancing, and performance management makes this single disk pool concept the right choice for most customers.

However, for many years, customers have found the need for additional storage granularity, including the need to sometimes isolate data into a separate disk pool. This is possible with User ASPs. User ASPs provide the same automation and ease-of-use benefits as the System ASP, but provide additional storage isolation when needed. With software level Version 5, IBM i takes this storage granularity option a huge step forward with the availability of *Independent Auxiliary Storage Pools (IASPs)*.

8.2 Boot from SAN and cloning

Traditionally, System i hosts have required the use of an internal disk as a boot drive or Load Source unit (LSU or LS). *Boot from SAN* support has been available since IBM i5/OS™ V5R3M5. IBM i Boot from SAN is supported on all types of external storage that attach to IBM i (natively or with Virtual I/O Server); this includes XIV storage.

For requirements for IBM i Boot from SAN with XIV see *IBM XIV Storage System with the Virtual I/O Server and IBM i,* REDP-4598.

Boot from SAN support enables IBM i customers to take advantage of Copy Services functions in XIV. These functions allow users to perform an instantaneous copy of the data held on XIV logical volumes. Therefore, when they have a system that only has external LUNs with no internal drives, they are able to create a clone of their IBM i system.

Important: In this book, a *clone* refers to a copy of an IBM i system that uses only external LUNs. Booting (or initial program loading) from SAN is therefore a prerequisite for this function.

Why consider cloning

By using the cloning capability, you can create a complete copy of your entire system in minutes. You can then use this copy in any way you want. For example, you could potentially use it to minimize your backup windows, or protect yourself from a failure during an upgrade, or maybe even use it as a fast way to provide yourself with a backup or test system. You can also use the remote copy of volumes for disaster recovery of your production system in case of failure or disaster at the primary site.

When you use cloning

Consider the following information when you use cloning:

- You need enough free capacity on your external storage unit to accommodate the clone. If Remote Mirroring is used you need enough bandwidth on the links between the XIV at the primary site and the XIV at the secondary site.
- The clone of a production system runs in a separate logical partition (LPAR) in POWER or Blade server, and therefore you need enough resources to accommodate it. In case of Remote Mirroring you need an LPAR in POWER server or Blade at the remote site where you will implement the clone.
- You should not attach a clone to your network until you have resolved any potential conflicts that the clone has with the parent system.

Note: Besides cloning, IBM i provides another way of using Copy Services on external storage: copying of an Independent Auxiliary Storage Pool (IASP) in a cluster. Implementations with IASP are not supported on XIV.

8.3 Our implementation

We set up the following environment on which to demonstrate XIV Copy functions with IBM i:

- System p6 model 570:
  - Two partitions with VIOS V2.2.0
  - An LPAR with IBM i V7.1 is connected to both VIOS with Virtual SCSI (VSCSI) adapters
- IBM XIV model 2810 connected to both VIOS with two 8 Gbps Fibre Channel adapters in each VIOS.
- Each connection between XIV and VIOS is done through one host port in XIV, each host port through a separate Storage Area Network (SAN).

Note: An advisable approach is to connect multiple host ports in XIV to each adapter in the host server; however, for the purpose of our example, we connected only one port in XIV to each VIOS.

- IBM i disk capacity in XIV: 8 * 137.4 GB volumes are connected to both VIOS.

  The volume capacity stated here is the *net* capacity available to IBM i. For more information about the XIV usable capacity for IBM i see *IBM XIV Storage System with the Virtual I/O Server and IBM i,* REDP-4598.

  - The corresponding disk units in each VIOS are mapped to the VSCSI adapter assigned to the IBM i partition.
  - Because the volumes are connected to IBM i through two VIOS, IBM i Multipath was automatically established for those volumes. As can be seen in Figure 8-2, the IBM i resource names for the XIV volumes starts with DPM which denotes that the disk units are in Multipath.
  - IBM i Boot from SAN is implemented on XIV.

Figure 8-2 shows the Display Disk Configuration Status in IBM i System Service Tools (SST).

```
                    Display Disk Configuration Status

           Serial                        Resource                    Hot Spare
ASP Unit   Number          Type Model    Name       Status           Protection
  1                                                 Unprotected
       1   Y37DQDZREGE6    6B22  050     DMP002     Configured            N
       2   Y33PKSV4ZE6A    6B22  050     DMP003     Configured            N
       3   YQ2MN79SN934    6B22  050     DMP015     Configured            N
       4   YGAZV3SLRQCM    6B22  050     DMP014     Configured            N
       5   YS9NR8ZRT74M    6B22  050     DMP007     Configured            N
       6   YH733AETK3YL    6B22  050     DMP005     Configured            N
       7   Y8NMB8T2W85D    6B22  050     DMP012     Configured            N
       8   YS7L4Z75EUEW    6B22  050     DMP010     Configured            N









Press Enter to continue.

F3=Exit        F5=Refresh         F9=Display disk unit details
F11=Disk configuration capacity   F12=Cancel
```

Figure 8-2 XIV volumes in IBM i Multipath

- Configuration for snapshots

  For the purpose of our experimentation, we used one IBM i LPAR for both production and backup. The LPAR was connected with two Virtual I/O Servers. Before initial program loading of the IBM i clone from snapshots, we unmapped the virtual disks from the production IBM i and we mapped the corresponding snapshot hdisks to the same IBM i LPAR, in each VIOS.

  Obviously, in real situations, you should use two IBM i LPARs (production LPAR and backup LPAR). The same two VIOS can be used to connect each production and backup LPAR. In each VIOS, the snapshots of production volumes will be mapped to the backup IBM i LPAR.

- Configuration for Remote Mirroring

  For the purpose of our experimentation, we used one IBM i LPAR for both production and disaster recovery. Before initial program loading of the IBM i clone from the Remote Mirror secondary volumes, we unmapped the virtual disks of production IBM i and we mapped the hdisks of mirrored secondary volumes to the same IBM i LPAR in each VIOS.

  Again, in real situations, you should use two IBM i LPARs (production LPAR and Disaster recovery LPAR), each of them in a different POWER server or blade server, and each connected with two different VIOS.

8.4 Snapshots with IBM i

Cloning a system from the snapshots can be employed in IBM i backup solutions. Saving of application libraries, objects, or an entire IBM i system to tape is done from the clone of a production system that resides in a separate logical partition (called a backup partition) in the POWER server. This solution brings many benefits, in particular those described in the next section.

As noted in 8.1.2, “Single-level storage” on page 218, IBM i data is kept in the main memory until it is swapped to disk as a result of a page fault. Before cloning the system with snapshots it is necessary to make sure that the data was flushed from memory to disk. Otherwise the backup system that is then started from snapshots would not be consistent (up-to-date) with the production system; even more important, the backup system would not use consistent data, which can cause the failure of initial program load (IPL).

Some IBM i customers prefer to power off their systems before creating or overwriting the snapshots, to make sure that the data is flushed to disk. Or, they force the IBM i system to a restricted state before creating snapshots.

However, in many IBM i centers it is difficult or impossible to power off the production system every day before taking backups from the snapshots. Instead, one can use the IBM i quiesce function provided in V6.1 and later. The function writes all pending changes to disk and suspends database activity within an auxiliary storage pool (ASP). The database activity remains suspended until a Resume is issued. This is known as *quiescing the ASP*.

When cloning the IBM i, you should use this function to quiesce the *SYSBAS* which means quiescing all ASPs except independent ASPs. If there are independent ASPs in your system, they should be varied-off before cloning. When using this function, set up the XIV volumes in a consistency group.

The details concerning both methods (powering down the IBM i and quiescing the ASP) are provided later in this section.

8.4.1 Solution benefits

Taking IBM i backups from a separate LPAR provides the following benefits to an IBM i center:

- The production application downtime is only while is necessary to power off the production partition, take a snapshot, or overwrite the snapshot of the production volumes, and start the production partition (IPL is normal). Usually, this time is much shorter than the downtime experienced when saving to tape without a Save While Active function. (The Save While Active function allows you to save IBM i objects to tape without the need to stop updates on these objects.)

  Save to tape is usually a part of batch job, the duration of which is critical for an IT center. This makes it even more important to minimize the production downtime during the save.
- The performance impact on the production application during the save to tape operation is minimal because it does not depend on IBM i resources in the production system.
- This solution can be implemented together with Backup, Recovery, and Media Services for IBM iSeries® (BRMS), an IBM i software solution for saving application data to tape.

8.4.2 Disk capacity for the snapshots

If the storage pool is about to become full because of redirect-on-write operations, the XIV Storage System automatically deletes a snapshot. IBM i deletion of a snapshot while the backup partition is running would cause a crash of the backup system. To avoid such a situation consider allocating enough space to the storage pool to accommodate snapshots for the time your backup LPAR is running.

Snapshots must have at least 34 GB allocated. Because the space needed depends on the size of LUNs and the location of write operations, the initial allocation should be a conservative estimate of about 80% of the source capacity to the snapshots. Then, monitor how snapshot space is growing during the backup. If snapshots do not use all the of the allocated capacity, you can adjust the snapshot capacity to a lower value.

For an explanation of how to monitor the snapshot capacity, see *IBM XIV Storage System Architecture and Implementation*, SG24-7659.

8.4.3 Power-down IBM i method

To clone IBM i using XIV snapshots, perform the following steps:

1. Power off the IBM i production system.
   a. Issue the `PWRDWNSYS` command (Figure 8-3). Specify to end the system using the `Controlled end time delay`.
   b. In the scenario with snapshots you do not want IBM i to restart immediately after shutdown, so specify `Restart option *NO`.

```
                        Power Down System (PWRDWNSYS)

Type choices, press Enter.

How to end . . . . . .   *CNTRLD       *CNTRLD, *IMMED
Controlled end delay time  . . .   10           Seconds, *NOLIMIT
Restart options:
  Restart after power down . . .   *NO          *NO, *YES
  Restart type . . . . . .   *IPLA         *IPLA, *SYS, *FULL
IPL source . . . . . .   *PANEL       *PANEL, A, B, D, *IMGCLG




Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display        F24=More keys
```

Figure 8-3 Power-down IBM i

After you confirm to power down the system, IBM i starts to shut down; you can follow the progress by observing SRC codes in the HMC of the POWER server or, as in our example, of the System p server.

After shut down, the system shows as `Not Activated` in the HMC, as can be seen in Figure 8-4.

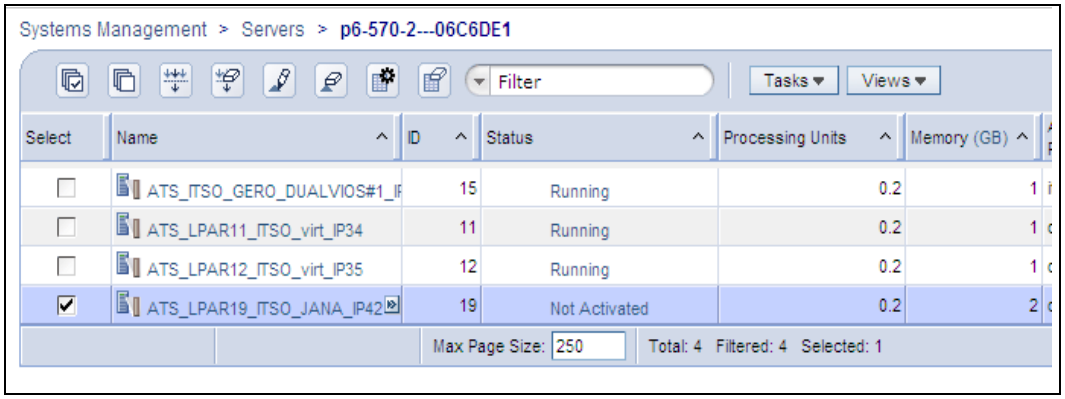


Figure 8-4 IBM i LPAR Not Activated

2. Create snapshots of IBM i volumes.

   You *create* the snapshot only the first time you execute this scenario. For subsequent executions, you can *overwrite* the snapshot.

   a. In the XIV GUI expand **Volumes** → **Volumes and Snapshots**, as shown in Figure 8-5.

Figure 8-5 XIV GUI Volumes and Snapshots

   b. In the Volumes and Snapshots panel, right-click each IBM i volume and click **Create Snapshot**. The snapshot volume is immediately created and shows in the XIV GUI. Notice that the snapshot volume has the same name as the original volume with suffix `snapshot` appended to it. The GUI also shows the date and time the snapshot was created. For details of how to create snapshots, see 2.2.1, “Creating a snapshot” on page 11.

   In everyday usage it is a good idea to overwrite the snapshots. You create the snapshot only the first time, then you overwrite it each time you need to take a new backup. The overwrite operation modifies the pointers to the snapshot data, therefore the snapshot appears as new. Storage that was allocated for the data changes between the volume and its snapshot is released.

   For details of how to overwrite snapshots, refer 2.2.5, “Overwriting snapshots” on page 18.

3. Unlock the snapshots.

   This action is needed only after you create snapshots. The created snapshots are locked, which means that a host server can only read data from them, but the data cannot be modified. For IBM i backup purposes the data on snapshots must be available for reads and writes, so it is necessary to unlock the volumes before using them for cloning IBM i. To unlock the snapshots use the Volumes and Snapshots window in XIV GUI, right-click each volume you want to unlock, and click **Unlock**.

After overwriting the snapshots, you do not need to unlock them again.

For details of how to overwrite snapshots, see 2.2.5, “Overwriting snapshots” on page 18.

4. Connect the snapshots to the backup IBM i LPAR.

   You must map the snapshot volumes to Virtual I/O Servers (VIOS) and map the corresponding virtual disks to IBM i adapters only the first time you use this approach. For subsequent executions, the existing mappings are used, and you just have to rediscover the devices in each VIOS with the **cfgdev** command. In each VIOS map the disk devices to the Virtual SCSI Server adapter to which the IBM i client adapter is assigned, using the **mkvdev** command:

   ```
   mkvdev -vdev hdisk16 -vadapter vhost0
   ```

   After the relevant disk devices are mapped to VSCSI adapters that connect to IBM i, they become part of the hardware configuration IBM i LPAR.

5. Start (by IPL) the IBM i backup system from snapshots.

   In the HMC of the POWER server IPL IBM i backup partition, select the LPAR and choose **Operations** → **Activate** from the pop-up menu, as shown in Figure 8-6.

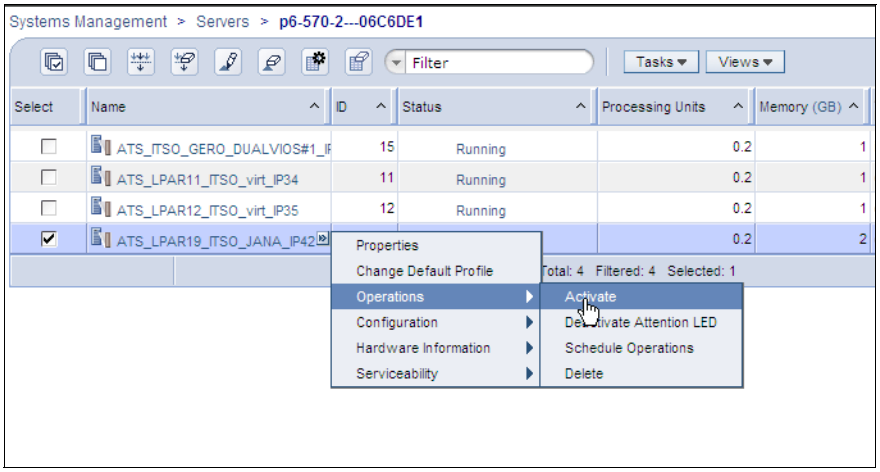


Figure 8-6 IPL of IBM i backup LPAR

The backup LPAR now hosts the clone of the production IBM i. Before using it for backups make sure that it is not connected to the same IP addresses and network attributes as the production system. For more information, see *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120.

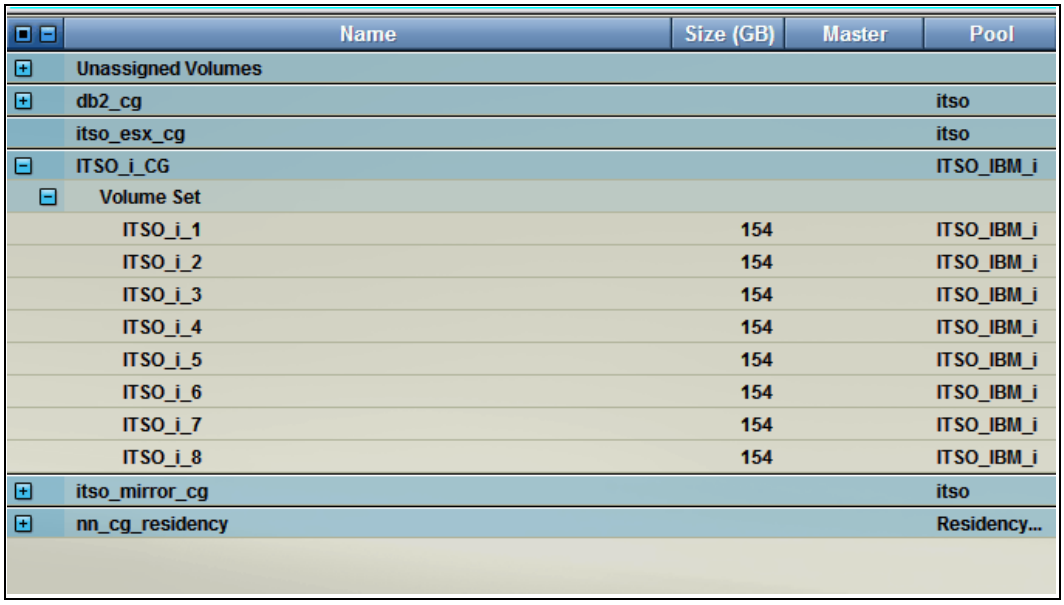
8.4.4 Quiescing IBM i and using snapshot consistency groups

To clone IBM i with XIV snapshots with consistency groups and the IBM i quiesce function, perform the following steps:

1. Create a consistency group and add IBM i volumes to it.

   For details of how to create the consistency group, see 2.3, “Snapshots consistency group” on page 24.

   The consistency group Diastolic used in our example is shown in Figure 8-7.



Name	Size (GB)	Master	Pool
Unassigned Volumes			
db2_cg			itso
itso_esx_cg			itso
ITSO_i_CG			ITSO_IBM_i
Volume Set			
ITSO_i_1	154		ITSO_IBM_i
ITSO_i_2	154		ITSO_IBM_i
ITSO_i_3	154		ITSO_IBM_i
ITSO_i_4	154		ITSO_IBM_i
ITSO_i_5	154		ITSO_IBM_i
ITSO_i_6	154		ITSO_IBM_i
ITSO_i_7	154		ITSO_IBM_i
ITSO_i_8	154		ITSO_IBM_i
itso_mirror_cg			itso
nn_cg_residency			Residency...

Figure 8-7 Volumes in consistency group

2. Quiesce the SYSBAS in IBM i and suspend transactions.

   To quiesce IBM i data to disk, use the IBM i command `CHGASPACT *SUSPEND`. Set the `Suspend Timeout` parameter to 30 seconds and `Suspend Timeout Action` to `*END`, as shown in Figure 8-8 on page 228.

   This causes the IBM i to flush as much transaction data as possible from memory to disk, then it waits for the specified time-out to get all current transactions to their next commit boundary and does not let them continue past that commit boundary. If the command succeeded after the time-out, the non-transaction operations are suspended and data that is non-pinned in the memory is flushed to disk.

   For detailed information about quescing data to disk with `CHGASPACT` see the following publications:

   – *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120,

   – *DS8000 Copy Services for IBM i with VIOS*, REDP-4584

   – *Implementing PowerHA for IBM i*, SG24-7405

   When the **`CHGASPACT`** command completes successfully, a message is indicates that the access to SYSBAS is suspended (Figure 8-9 on page 228).

```
                      Change ASP Activity (CHGASPACT)

Type choices, press Enter.

ASP device . . . . . > *SYSBAS       Name, *SYSBAS
Option . . . . . > *SUSPEND      *SUSPEND, *RESUME, *FRCWRT
Suspend timeout  . . . . . .   30           Number
Suspend timeout action . . . . .   *end         *CONT, *END









                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 8-8 Quiesce data to disk

```
MAIN                           IBM i Main Menu
                                                        System:   T00C6DE1
 Select one of the following:

      1. User tasks
      2. Office tasks
      3. General system tasks
      4. Files, libraries, and folders
      5. Programming
      6. Communications
      7. Define or change the system
      8. Problem handling
      9. Display a menu
     10. Information Assistant options
     11. IBM i Access tasks

     90. Sign off

 Selection or command
 ===>

 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
 F23=Set initial menu
 Access to ASP *SYSBAS is suspended.
```

Figure 8-9 Access to SYSBAS suspended

3. Create snapshots of IBM i volumes in the consistency group.

   You *create* the snapshot only the first time this scenario is executed. For subsequent executions, you can just *overwrite* the snapshot.

   In the Volumes and Snapshots panel, right-click each IBM i volume and click **Create Snapshot**. The snapshot volume is immediately created and shows in the XIV GUI. Notice

that the snapshot volume has the same name as the original volume with suffix "snapshot" appended to it. The GUI also shows the date and time the snapshot was created. For details of how to create snapshots, see 2.2.1, "Creating a snapshot" on page 11.

In everyday usage it is a good idea to overwrite the snapshots. You create the snapshot only the first time, then you overwrite it every time you need to take a new backup. overwrite operation modifies the pointers to the snapshot data, therefore the snapshot appears as new. Storage that was allocated for the data changes between the volume and its snapshot is released.

For details of how to overwrite snapshots, see 2.2.5, "Overwriting snapshots" on page 18.

4. Resume transactions in IBM i.

   After snapshots are created, resume the transactions in IBM i with the **CHGASPACT** command and ***RESUME** option, as shown in Figure 8-10.

```
                      Change ASP Activity (CHGASPACT)

Type choices, press Enter.

ASP device . . . . . .   *sysbas       Name, *SYSBAS
Option . . . . . .       *resume       *SUSPEND, *RESUME, *FRCWRT




                                                                  Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 8-10 Resume transactions in IBM i

   Look for the IBM i message `Access to ASP *SYSBAS successfully resumed`, to be sure that the command was successfully performed.

5. Unlock the snapshots in the consistency group.

   This action is needed only after you create snapshots. The created snapshots are locked, which means that a host server can only read data from them, but their data cannot be modified. Before starting (IPL) IBM i from the snapshots you have to unlock them to make them accessible for writes also. For this, use the Consistency Groups panel in the XIV GUI, right-click the snapshot group and select **Unlock** from the pop-up menu.

   After overwriting the snapshots, you do not need to unlock them again.

   For details about overwriting snapshots, see 2.2.5, "Overwriting snapshots" on page 18.

6. Connect the snapshots to the backup LPAR.

   Map the snapshot volumes to Virtual I/O Servers and map the corresponding virtual disks to IBM i adapters only the first time you use this solution. For subsequent operations the existing mappings are used; you just have to rediscover the devices in each VIOS using the **cfgdev** command.

Use these steps to connect the snapshots in the snapshot group to a backup partition:

a. In the Consistency Groups panel select the snapshots in the snapshot group, right-click any of them and select **Map selected volumes**, as shown in Figure 8-11.

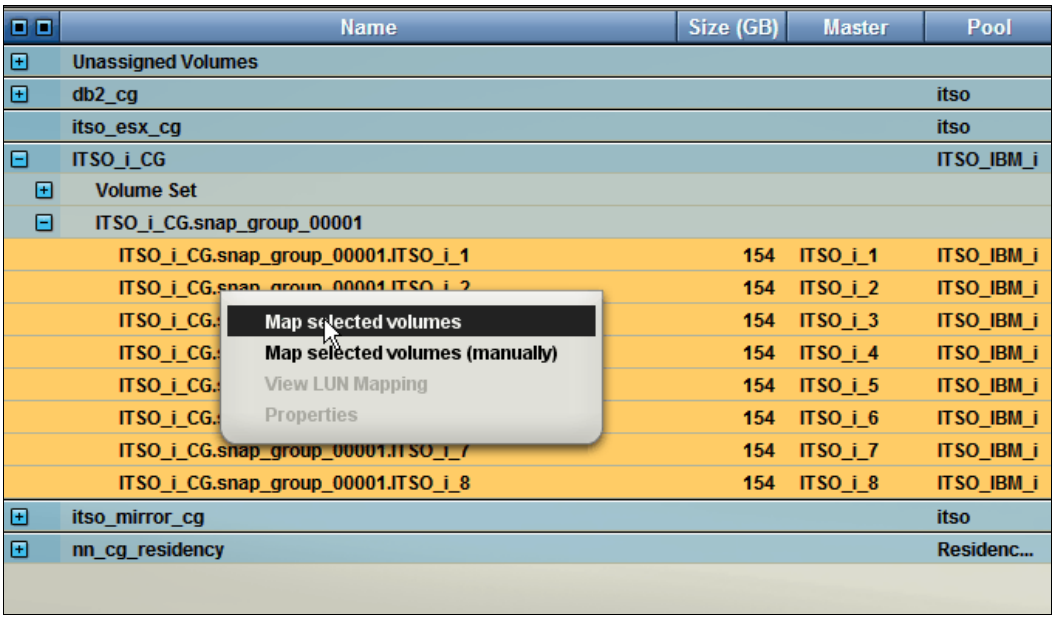


Figure 8-11 Map the snapshot group

b. In the next panel, select the host or cluster of hosts to map the snapshots to. In our example, we mapped them to the two Virtual I/O Servers that connect to IBM i LPAR. Here the term *cluster* refers only to the host names and their WWPNs in XIV; it does not mean that Virtual I/O Servers are in an AIX cluster. In each VIOS, rediscover the mapped volumes and map the corresponding devices to the VSCSI adapters in IBM i.

7. Start (with IPL) the IBM i backup system from snapshots. IPL the backup LPAR as described in step 5 on page 226. When you power off the production system before taking snapshots, the IPL of the backup system shows the previous system end as *normal*; quiescing data to disk before taking snapshots, the IPL of the backup LPAR shows the previous system end as *abnormal*, as shown in Figure 8-12.

```
                    Operating System IPL in Progress
                                                          10/01/10  12:57:24
 IPL:
   Type  . . . . . . . . . . . . . . . :   Attended
   Start date and time . . . . . . . . :   10/01/10  12:56:17
   Previous system end . . . . . . . . :   Abnormal
   Current step / total  . . . . . . . :   35   49
   Reference code detail . . . . . . . :   C900 2AA3  20 AC 0400

   IPL step                                    Time Elapsed   Time Remaining
   Commit recovery                               00:00:05
   Journal recovery - 2                          00:00:00
 > Database recovery - 2                         00:00:00
   Damage notification end - 2
   Spool initialization - 2
```

Figure 8-12 Abnormal IPL after quiesce data

8.4.5 Automation of the solution with snapshots

Many IBM i environments require their backup solution with snapshots to be fully automated so it can be run with a single command, or even scheduled for a certain time in day.

Automation for such a scenario can be provided in an AIX or Linux system, using XCLI scripts to manage snapshots, and Secure Shell (SSH) commands to IBM i LPAR and the HMC.

Note: IBM i must be set up for receiving SSH commands. For instructions see the paper *Securing Communications with OpenSSH on IBM i5/OS*, REDP-4163.

In our exercise, we used the AIX script to do the following procedures:

1. Send an SSH command to the production IBM i to suspend transactions and quiesce SYSBAS data to disk.

   ```
   CHGASPACT ASPDEV(*SYSBAS) OPTION(*SUSPEND) SSPTIMO(30)
   ```

2. Send an XCLI command to overwrite the snapshot group, or create a new one if there is not one. We used these XCLI commands:

   ```
   cg_snapshots_create cg=CG_NAME snap_group=SNAP_NAME
   cg_snapshots_create cg=CG_NAME overwrite=SNAP_NAME
   ```

3. Unlock the snapshot group with this XCLI command:

   ```
   snap_group_unlock snap_group=SNAP_NAME
   ```

4. Resume suspended transactions on the production IBM i:

   ```
   CHGASPACT ASPDEV(*SYSBAS) OPTION(*RESUME)
   ```

5. To rediscover the snapshot devices, send to each VIOS this SSH command:

   ```
   ioscli cfgdev
   ```

6. To start the backup LPAR that is connected to snapshot volumes, send the following SSH command to the IBM POWER HMC to start the backup LPAR that is connected to snapshot volumes:

   ```
   chsysstate -m hmc_ibmi_hw -r lpar -o on -n hmc_ibmi_name -f hmc_ibmi_prof
   ```

The script we used is shown in the Example 8-1.

Example 8-1 Automating the backup solution with snapshots

```
#!/bin/ksh
ssh_ibmi=qsecofr@1.2.3.5

XCLI=/usr/local/XIVGUI/xcli
XCLIUSER=itso
XCLIPASS=password
XIVIP=1.2.3.4
  CG_NAME=ITSO_i_CG
SNAP_NAME=ITSO_jj_snap

ssh_hmc=hscroot@1.2.3.4
hmc_ibmi_name=IBMI_BACKUP
hmc_ibmi_prof=default_profile
hmc_ibmi_hw=power570

ssh_vios1=padmin@1.2.3.6
ssh_vios2=padmin@1.2.3.7
```

```
# Suspend IO activity
ssh ${ssh_ibmi} 'system "CHGASPACT ASPDEV(*SYSBAS) OPTION(*SUSPEND) SSPTIMO(30)"'

# Check, whether snapshot already exists and can be overwritten
# otherwise create a new one and unlock it (it's locked by default)
${XCLI} -u ${XCLIUSER} -p ${XCLIPASS} -m ${XIVIP} -s snap_group_list
snap_group=${SNAP_NAME} >/dev/null 2>&1
RET=$?
# is there a snapshot for this cg?
if [ $RET -ne 0 ]; then
   # there is none, create one
   ${XCLI} -u ${XCLIUSER} -p ${XCLIPASS} -m ${XIVIP} cg_snapshots_create cg=${CG_NAME}
snap_group=${SNAP_NAME}
   # and unlock it
   ${XCLI} -u ${XCLIUSER} -p ${XCLIPASS} -m ${XIVIP} snap_group_unlock
snap_group=${SNAP_NAME}
fi

# overwrite snapshot
${XCLI} -u ${XCLIUSER} -p ${XCLIPASS} -m ${XIVIP} cg_snapshots_create cg=${CG_NAME}
overwrite=${SNAP_NAME}

# resume IO activity
ssh ${ssh_ibmi} 'system "CHGASPACT ASPDEV(*SYSBAS) OPTION(*RESUME)"'

# rediscover devices
ssh ${ssh_vios1} 'ioscli cfgdev'
ssh ${ssh_vios2} 'ioscli cfgdev'

# start the backup partition
ssh ${ssh_hmc} "chsysstate -m ${hmc_ibmi_hw} -r lpar -o on -n ${hmc_ibmi_name} -f
${hmc_ibmi_prof}"
```

In the backup IBM i LPAR you must change the IP addresses and network attributes so that they do not collide with the ones in the production LPAR. For this you can use the *startup* CL program in the backup IBM i; this is explained in detail in *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120.

You might also want to automate saving to tape in BRMS, by scheduling the save in BRMS. After the save the library QUSRBRM must be transferred to the production system.

8.5 Synchronous Remote Mirroring with IBM i

Synchronous Remote Mirroring used with IBM i boot from SAN provides the functionality for cloning a production IBM i system at a remote site. The remote clone can be used to continue production workload during planned outages or in the case of a disaster at the local site, and therefore provides a disaster recovery (DR) solution for an IBM i center.

A standby IBM i LPAR is needed at the DR site. After the switchover of mirrored volumes during planned or un-planned outages, perform an IPL of the standby partition from the mirrored volumes at the DR site. This ensures continuation of the production workload in the clone.

Typically, synchronous mirroring is used for DR sites located at shorter distances, and for IBM i centers that require a near zero Recovery Point Objective (RPO). On the other hand,

clients that use DR centers located at long distance and who can cope with a little longer RPO might rather implement Asynchronous Remote Mirroring.

Use consistency groups with synchronous mirroring for IBM i to simplify management of the solution and to provide consistent data at the DR site after re-snychronization following a link failure.

8.5.1 Solution benefits

Synchronous Remote Mirroring with IBM i offers the following major benefits:

- It can be implemented without any updates or changes to the production IBM i.
- The solution does not require any special maintenance on the production or standby system partition. Practically, the only required task is to set up the synchronous mirroring for all the volumes making up the entire partition disk space. After this is done, no further actions are required.
- Because synchronous mirroring is completely handled by the XIV system, this scenario does not use any processor or memory resources from either the production or remote IBM i partitions. This is different from other IBM i replication solutions, which require some processor resources from the production and recovery partitions.

8.5.2 Planning the bandwidth for Remote Mirroring links

In addition to the points specified in 4.6, “Planning” on page 97, an important step is to provide enough bandwidth for the connection links between the primary and secondary XIV used for IBM i mirroring. Proceed as follows to determine the necessary bandwidth (MBps):

1. Collect IBM i performance data. Do the collection over at least a one-week period, and if applicable, during heavy workload such as when running end-of-month jobs.

   For more information about IBM performance data collection see *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120.
2. Multiply the writes per second by the reported transfer size to get the write rate (MBps) for the entire period over which performance data was collected.
3. Look for the highest reported write rate. Size the Remote Mirroring connection so that the bandwidth can accommodate the highest write rate.

8.5.3 Setting up synchronous Remote Mirroring for IBM i

Use the following steps to set up synchronous Remote Mirroring with consistency group for IBM i volumes:

1. Configure Remote Mirroring, as described in 4.11, “Using the GUI or XCLI for remote mirroring actions” on page 99.
2. Establish and activate synchronous mirroring for IBM i volumes, as described in 4.12, “Configuring remote mirroring” on page 112.
3. Activate the mirroring pairs, as described in 5.1, “Synchronous mirroring considerations” on page 114.

   Figure 8-13 on page 234 shows the IBM i mirroring pairs used in our scenario during the initial synchronization. Some of the mirroring pairs are already in synchronized status, whereas some of them are still in initialization state with a reported percentage synchronized volume.

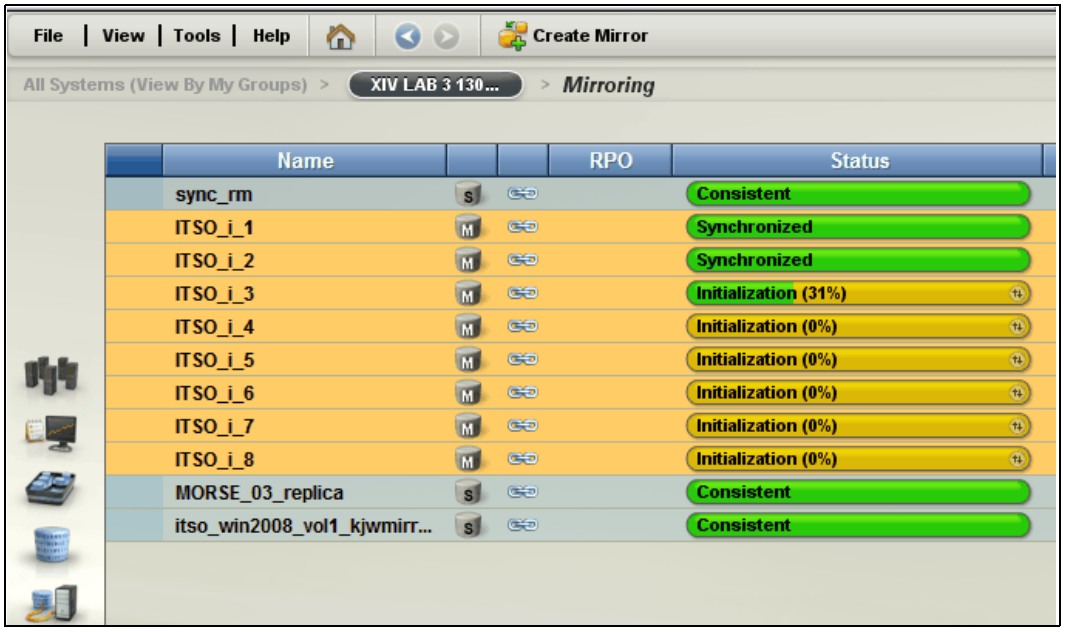


Figure 8-13 Synchronizing of IBM i mirrored pairs

4. Create a mirror consistency group and activate mirroring for the CG on *both primary and secondary* XIV systems.

   Setting a consistency group to be mirrored is done by first creating a consistency group, then setting it to be mirrored, and only then populating it with volumes. A consistency group must be created at the primary XIV and a corresponding consistency group at the secondary XIV. The names of the consistency groups can be different.

   To activate the mirror for the CG, in the XIV GUI Consistency Groups panel for the primary XIV, right-click the created consistency group and select **Create Mirror**.

   For details, see 5.3.2, “Using the GUI for CG mirroring setup” on page 122.

5. Add the mirrored volumes to the consistency group.

   Note: When adding the mirrored volumes to the consistency group, all volumes and the CG must have the same status. Therefore the mirrored volumes should be synchronized before you add them to the consistency group, and the CG should be activated, so that all of them have status Synchronized.

   In the primary XIV system select the IBM i mirrored volumes, right-click and select **Add to Consistency Group**.

Figure 8-14 shows the consistency group in synchronized status for our scenario.

Figure 8-14 CG in synchronized status

8.5.4 Scenario for planned outages

Many IBM i IT centers minimize the downtime during planned outages (such as for server hardware maintenance or installing program fixes) by switching their production workload to the DR site during the outage.

Note: *Switching* mirroring roles is suitable for planned outages during which the IBM i system is powered-down. For planned outages with IBM i running, *changing* the mirroring roles is more appropriate.

With synchronous mirroring, use the following steps to switch to the DR site for planned outages:

1. Power off the production IBM i system as described in step 1 of 8.4.3, “Power-down IBM i method” on page 224.
2. Switch the roles of mirrored XIV volumes.

   To do this, use the GUI for the *primary* XIV, and in the Mirroring window right-click the consistency group that has the IBM i mirrored volumes, then select **Switch Roles**, as is shown in Figure 8-15 on page 236.

Figure 8-15 Switch the roles of mirrored volumes for IBM i

Confirm switching the roles in your consistency group by clicking **OK** in the Switch Roles pop-up dialog. After the switch is done, the roles of mirrored volumes are reversed: the IBM i mirroring consistency group on the primary XIV is now the slave and the consistency group on the secondary XIV is now the master. This is shown in Figure 8-16, which also shows that the status of CG at the primary site now has the Consistent status; at the secondary site, the status is Synchronized.

Figure 8-16 Mirrored CG after switching the roles

3. Make the mirrored secondary volumes available to the standby IBM i.

   You might want to have the secondary volumes mapped to the adapters in Virtual I/O Servers, and their corresponding hdisks mapped to virtual adapters in the standby IBM i at all times. In that case you need to do this setup only the first time you recover from mirrored volumes; from then on, the devices will use the existing mapping so you just have to rediscover them.

   The assumption is that the following steps are done at the DR site:

   – The physical connection of XIV to the adapters in Virtual I/O Servers has been made.
   – The hosts and optionally clusters are defined in XIV.
   – The ports of adapters in Virtual I/O Servers are added to the hosts in XIV.

To connect the mirrored volumes to the DR IBM i system, do the following steps:

a. Map the secondary volumes to the WWPNs of adapters (as described in step 6 on page 229).

b. In each VIOS, discover the mapped volumes using the **cfgdev** command.

c. In each VIOS, map the devices (hdisks) that correspond to the secondary volumes to the virtual adapters in the standby IBM i, (as described in step 4 on page 226).

4. Perform an IPL of the standby IBM i LPAR.

   Perform IPL of the disaster recovery IBM i LPAR (as described in step 5 on page 226). Because the production IBM i was powered-down, the IPL of its clone at the DR site is *normal* (previous system shutdown was normal).

   If both the production and the DR IBM i are in the same IP network, it is necessary to change the IP addresses and network attributes of the clone at the DR site. For more information about this see *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120.

After the production site is available again, you can switch back to the regular production site, by executing the following steps:

1. Power off the DR IBM i system (as described in step 1 on page 224).
2. Switch the mirroring roles of XIV volumes (as described in step 2 on page 235).

> **Note:** When switching back to the production site you must initiate the role switching on the *secondary (DR)* XIV, because role switching must be done on the master peer.

3. In each VIOS at the primary site, rediscover the mirrored primary volumes by issuing the **cfgdev** command.
4. Perform an IPL of the production IBM i LPAR (as described in step 5 on page 226). Because the DR IBM i was powered off, the IPL of its clone in the production site is now *normal* (previous system shutdown was normal).

8.5.5 Scenario for unplanned outages

In case of failure at the production IBM i, caused by any unplanned outage on the primary XIV system or a disaster, recover your IBM i at the DR site from mirrored secondary volumes.

For our scenario, we simulated failure of the production IBM i by unmapping the virtual disks from the IBM i virtual adapter in each VIOS so that IBM i missed the disks and entered the DASD attention status. The SRC code showing this status is in Figure 8-17.

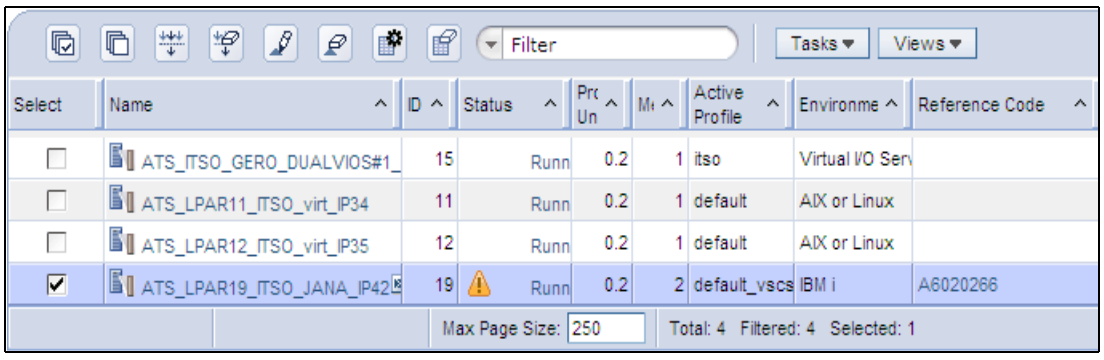


Figure 8-17 IBM i DASD attention status at disaster

Recover from this disaster by using the following steps:

1. Change the peer roles at the secondary site.

   Change the roles of secondary mirrored volumes from slave to master as follows:

   a. In the GUI of the secondary XIV, select **Remote** → **Mirroring**.

   b. Right-click the mirroring consistency group that contains the IBM i volumes and select **Change Role**. Confirm changing the role of the slave peer to master.

   Changing the roles stops mirroring. The mirror status is shown as Inactive on the secondary site, and the secondary peer becomes master. The primary peer keeps the master role also, and the mirroring status on the primary site shows as Synchronized. This can be seen in Figure 8-18, which shows the secondary IBM i consistency group after changing the roles, and in Figure 8-19 showing the primary peer.

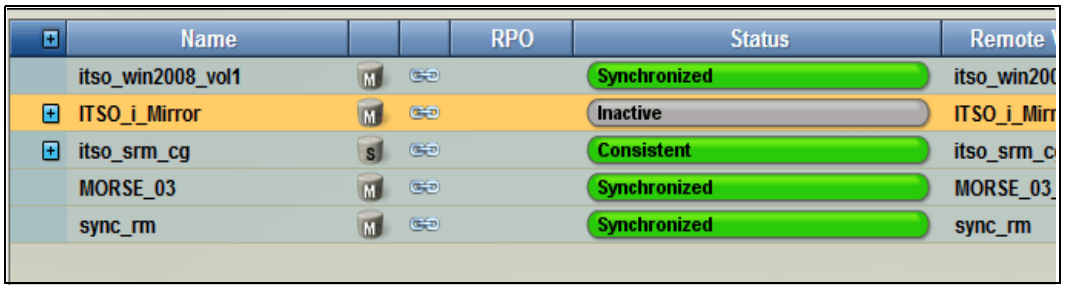


Figure 8-18 Secondary peer after changing the role

Figure 8-19 Primary peer after changing the role

2. Make the secondary volumes available to the standby IBM i.

   We assume that the physical connections from XIV to the POWER server at the DR site are already established.

   The following steps are required to make the secondary mirrored volumes available to IBM i at the DR site:

   a. In the secondary XIV, map the mirrored IBM i volumes to the adapters in VIOS (as described in step 4 on page 226).

   b. In each VIOS in POWER server at the DR site, use the **`cfgdev`** command to rediscover the secondary mirrored volumes.

   c. In each VIOS, map the devices that correspond to XIV secondary volumes to virtual host adapters for IBM i (as described in step 4 on page 226).

   You might want to keep the mappings of secondary volumes in XIV and in VIOS. In this case, the only step necessary is to rediscover the volumes in VIOS with **`cfgdev`** command.

3. IPL the IBM i at the DR site.

   Do an IPL of the standby IBM i LPAR at the DR site (as described in step 5 on page 226). IPL is *abnormal* (the previous system termination was abnormal) as shown in Figure 8-20. After recovery there might be damaged objects in the IBM i since the production system suffered a disaster. They are reported by operator messages and usually can be fixed by appropriate procedures in IBM i. The message about damaged objects in our example is shown in Figure 8-21 on page 240.

```
              Licensed Internal Code IPL in Progress
                                                          10/11/10  12:09:53
IPL:
  Type  . . . . . . . . . . . . . . . :   Attended
  Start date and time . . . . . . . . :   10/11/10  12:09:43
  Previous system end . . . . . . . . :   Abnormal
  Current step / total  . . . . . . . :      10     16
  Reference code detail . . . . . . . :   C6004057

IPL step                                 Time Elapsed   Time Remaining
 Journal Recovery                           00:00:01        00:00:00
 IFS Initialization                         00:00:01        00:00:00
>Data Base Recovery
 Journal Synchronization
 Commit Recovery

Item:
  Current / Total . . . . . . . :

Sub Item:
  Identifier  . . . . . . . . . :
  Current / Total . . . . . . . :
```

Figure 8-20 IPL of standby LPAR after disaster

```
                              Display Messages
                                                          System:   T00C6DE1
Queue . . . . . :   QSYSOPR               Program . . . . :   *DSPMSG
  Library . . . :     QSYS                   Library . . . :
Severity  . . . :   60                     Delivery  . . . :   *BREAK

Type reply (if required), press Enter.
  Subsystem QBASE active when system ended.
  Subsystem QSYSWRK active when system ended.
  Subsystem QSERVER active when system ended.
  Subsystem QUSRWRK active when system ended.
  Subsystem QSPL active when system ended.
  Subsystem QHTTPSVR active when system ended.
  455.61M of 490.66M for shared pool *INTERACT allocated.
  Damaged object found.



                                                                      Bottom
F3=Exit              F11=Remove a message                 F12=Cancel
F13=Remove all       F16=Remove all except unanswered    F24=More keys
```

Figure 8-21 Damaged object in IBM i after disaster recovery

If both production and DR IBM i are in the same IP network it is necessary to change the IP addresses and network attributes of the clone at the DR site. For more information, see *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120.

When the production site is back, failover to the normal production system as follows:

1. Change the role of primary peer to slave.

   On the primary XIV, select **Remote** → **Mirroring** in the GUI, right-click the consistency group of IBM i volumes, and select **Deactivate** from the pop-up menu, then right-click again and select **Change Role**. Confirm the change of the peer role from master to slave.

   Now the mirroring is still inactive, and the primary peer became the slave, so the scenario is prepared for mirroring from the DR site to the production site. The primary peer status is shown in Figure 8-22.

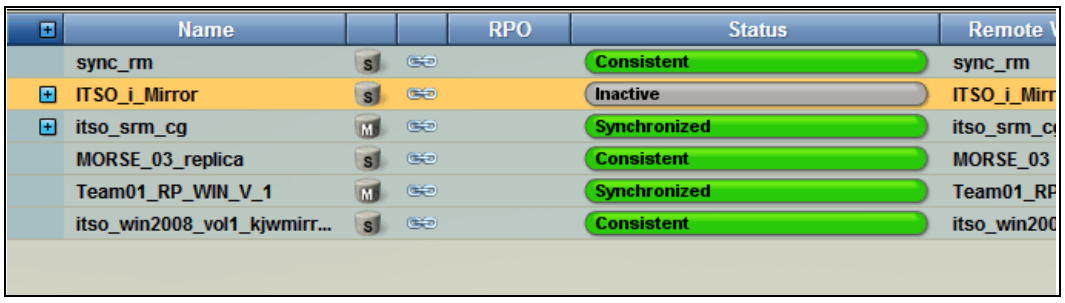


Figure 8-22 Primary peer after changing the roles

2. Activate the mirroring.

   In the GUI of the secondary XIV, select **Remote** → **Mirroring**, right-click the consistency group for IBM i volumes, and select **Activate**. Now the mirroring is started in the direction

from secondary to primary peer. At this point only the changes made on the DR IBM i system during the outage need to be synchronized, and the mirror synchronization typically takes little time.

After the mirroring is synchronized, do the following steps:

1. Power off the DR IBM i.

   Power off the IBM i at the DR site (as described in step 1 of 8.4.3, “Power-down IBM i method” on page 224).

2. Switch peer roles.

   On the secondary XIV, switch the mirroring roles of volumes (as described in step 2 on page 235).

3. Rediscover primary volumes in VIOS.

   In each VIOS on the primary site rediscover the mirrored primary volumes by issuing a **cfgdev** command.

4. Do an IPL of the production IBM i.

   Perform an IPL of the production IBM i LPAR (as described in step 5 on page 226).

8.6 Asynchronous Remote Mirroring with IBM i

This section describes Asynchronous Remote Mirroring of the local IBM i partition disk space. This solution provides continuous availability with a recovery site located at a long distance while minimizing performance impact on production.

In this solution, the entire disk space of the production IBM i LPAR resides on the XIV to allow boot from SAN. Asynchronous Remote Mirroring for all XIV volumes belonging to the production partition is established with another XIV located at the remote site.

In case of an outage at the production site a remote standby IBM i LPAR takes over the production workload with the capability to IPL from Asynchronous Remote Mirroring secondary volumes.

Because of the XIV Asynchronous Remote Mirroring design, the impact on production performance is minimal; however, the recovered data at the remote site is typically lagging production data because of the asynchronous nature, although usually just slightly behind. For more information about the XIV Asynchronous Remote Mirroring design and implementation, see Chapter 4, “Remote mirroring” on page 53.

8.6.1 Benefits of asynchronous Remote Mirroring

Solutions with asynchronous mirroring provide several significant benefits to an IBM i center:

- The solution provides replication of production data over long distances while minimizing production performance impact.
- The solution does not require any special maintenance on the production or standby partition. Practically, the only required task is to set up Asynchronous mirroring for the entire IBM i disk space.
- Because asynchronous mirroring is entirely driven by the XIV storage systems, this solution does not use any processor or memory resources from the IBM i production and remote partitions. This is different from other IBM i replication solutions, which use some of the production and recovery partitions’ resources.

8.6.2 Setting up asynchronous Remote Mirroring for IBM i

The following steps are used to set up asynchronous remote mirroring with consistency group for IBM i volumes:

1. Configure Remote Mirroring as is described in 4.11, “Using the GUI or XCLI for remote mirroring actions” on page 99.
2. Establish and activate asynchronous mirroring for IBM i volumes.
   a. To establish the asynchronous mirroring on IBM i volumes, using the GUI on the primary XIV, select **Volumes** → **Volumes and Snapshots**. Right-click each volume to mirror and select **Create Mirror** from the pop-up menu. In the Create Mirror window, specify `Synch Type Asynch`, specify the target XIV system and the slave volume to mirror to, desired RPO, and schedule management `XIV Internal`.

      For more information about establishing Asynchronous mirroring see 6.1, “Asynchronous mirroring configuration” on page 150.

   b. To activate asynchronous mirroring on IBM i volumes, using the GUI on the primary XIV, select **Remote** → **Mirroring**. Highlight the volumes to mirror and select **Activate** from the pop-up window. After activating, the initial synchronization of mirroring is performed.

      Figure 8-23 shows the IBM i volumes during initial synchronization: Some are already in the RPO OK status, one is in RPO lagging status, and several are not yet synchronized.

Figure 8-23 Initial synchronization of Asynchronous mirroring for IBM i volumes

3. Create a consistency group for mirroring on *both* the primary and secondary XIV systems, and activate mirroring on the CG (as described in step 4 on page 234).

   When activating the asynchronous mirroring for the CG, you must select the same options selected when activating the mirroring for the volumes.

   Before adding the volumes to the consistency group, the mirroring on all CG and the volumes must be in the same status. Figure 8-24 on page 243 shows the mirrored volumes and the CG before adding the volumes to CG in our example. The status of all of them is RPO OK.

Figure 8-24 Status before adding volumes to CG

4. Add the mirrored IBM i volumes to the consistency group (as described in step 5 on page 234).

8.6.3 Scenario for planned outages and disasters

For our scenario, we simulated the failure of the production IBM i by unmapping the virtual disks from the IBM i virtual adapter in each VIOS, so that IBM i misses the disks and entered the DASD attention status.

When you need to switch to the DR site for planned outages or as a result of a disaster, perform the following steps:

1. Change the role of the secondary peer from slave to master.

   Select **Remote** → **Mirroring** in the GUI for the secondary XIV. Right-click the mirrored consistency group and select **Change Role**. Confirm changing the role of the slave peer to master.

2. Make the mirrored secondary volumes available to the standby IBM i.

   We assume that the physical connections from XIV to the POWER server on the DR site are already established at this point. Rediscover the XIV volumes in each VIOS with the **`cfgdev`** command, then map them to the virtual adapter of IBM i (as described in step 3 on page 236.

3. IPL the IBM i and continue the production workload at the DR site (as described in step 3 on page 239).

After the primary site is available again, do the following steps:

1. Change the role of the primary peer from master to slave.

   On the primary XIV, select **Remote** → **Mirroring** in the GUI, right-click the consistency group of IBM i volumes, and select **Deactivate** from the pop-up menu. Right-click again and select **Change Role**. Confirm changing the peer role from master to slave.

2. Re-activate the asynchronous mirroring from the secondary peer to the primary peer.

   In the GUI of the secondary XIV go to **Remote** → **Mirroring**, right-click the consistency group for IBM i volumes, and select **Activate**. Now the mirroring is started in the direction

from the secondary to primary peer. At this point only the changes made on the DR IBM i system during the outage need to be synchronized, so the synchronization of mirroring typically takes little time.

In case the primary mirrored volumes do not exist anymore after the primary site is available again, you have to delete the mirroring in the XIV on the DR site. Then, establish the mirroring again with the primary peer on the DR site, and activate it.

3. Power off the DR IBM i.

   After the mirroring is synchronized and before switching back to the production site, power off the DR IBM i LPAR so that all data is flushed to disk on the DR site. Power off the DR IBM i (as described in step 1 on page 224).

4. Change the role of the primary peer from slave to master.
5. Change the role of the secondary peer from master to slave.
6. Activate mirroring.
7. Do an IPL of the production IBM i and continue the production workload.

   Do an IPL of the production IBM i LPAR (as described in step 7 on page 230). When the system is running, the production workload can resume on the primary site.

9

Data migration

The XIV storage system software includes, with no extra charge, a powerful data migration tool. It is used to help customers with the task that every Storage Administrator must confront when a new storage device is brought in to replace old storage. The XIV Data Migration Utility (XDMU) can migrate data from almost any storage system to the XIV Storage System.

During the migration initiation, hosts are offline for only a short time before being connected to the XIV. The original LUNs are then allocated to the XIV instead of the server and are then natively presented again to the host through the XIV. Meanwhile the data is transparently migrated in the background, in a controlled fashion.

This chapter includes usage examples and troubleshooting information.

© Copyright IBM Corp. 2014. All rights reserved.

9.1 Overview

Customers have a need for seamless data migration whenever their storage environment changes. Always avoid or minimize disruption to your business applications if possible. Although many options are available for migrating data from one storage system to another, the XIV Storage System includes a data migration tool to help more easily with the movement of data from an existing storage system to the XIV Storage System. This feature enables the production environment to continue functioning during the data transfer with only a short period of down time for your business applications. Figure 9-1 presents a high level view of a sample data migration environment.

For data migrations between XIV Storage Systems, the advice is to perform the migration by using synchronous remote mirroring (see Chapter 5, “Synchronous Remote Mirroring” on page 113) or by using asynchronous remote mirroring (see Chapter 6, “Asynchronous remote mirroring” on page 149).

By using XIV replication to migrate from XIV Generation 2 to Gen3, all data will be migrated (through replication) to the new XIV before the server is moved to the new XIV. The server outage is at the end of the process after the data is migrated (replicated) to the new XIV system. This method mitigates any issues that might arise if a failure occurs during the process that affects the paths (FC and iSCSI) between the old and new XIVs. It also minimizes any latency that might be introduced through the standard XIV Data Migration Utility. As with any migration, the best approach is to subdivide the process into manageable tasks. Moving all the data at once is not necessary and not recommended. The data is migrated through replication on a LUN-by-LUN basis.

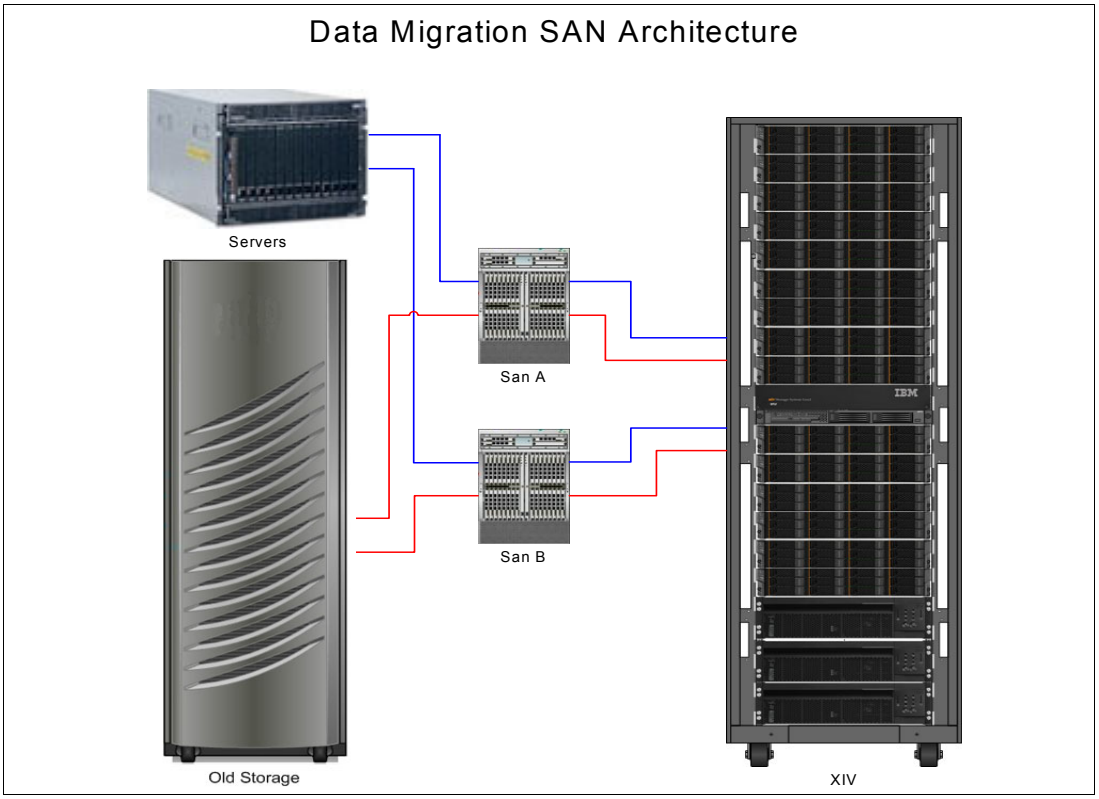


Figure 9-1 Data migration simple view

Note: Using asynchronous remote copy requires space for snapshots that are part of the replication architecture (see Chapter 6, “Asynchronous remote mirroring” on page 149).

The IBM XIV Data Migration Utility offers a seamless data transfer for the following reasons:

- Requires only a short outage to switch LUN ownership. This enables the immediate connection of a host server to the XIV Storage System, providing the user with direct access to all the data before it has been copied to the XIV Storage System.
- Synchronizes data between the two storage systems using transparent copying to the XIV Storage System as a background process with minimal performance impact.
- Enables a data synchronization rate that can be tuned through the XCLI interface without any impact to the server.
- Supports data migration from most storage vendors.
- Can be used with Fibre Channel or iSCSI.
- Can be used to migrate SAN boot volumes (OS-dependent).

AIX: For AIX, the preference is to use the built-in utilities (**`alt_disk_copy`** or **`migratepv`**) to migrate rootvg (SAN Boot Device)

XIV manages the data migration by simulating host behavior. When connected to the source storage system containing the source data, XIV looks and behaves like a SCSI initiator (Linux host/server). After the connection is established, the storage device containing the source data believes that it is receiving read or write requests from a Linux host, when in fact it is the XIV Storage System doing a block-by-block copy of the data, which the XIV is then writing onto an XIV volume.

The migration process allows for the XIV to perform a copy process of the original data in the background. While this is happening the host server is connected to the XIV and it can access its storage without disruption. XIV will handle all of the server’s read and write requests. If the data that the server requires is not on the XIV, the copy process retrieves the data from the source storage system. The entire data transfer is transparent to the host server and the data is available for immediate access.

Be sure that the connections between the two storage systems remain intact during the entire migration process. If at any time during the migration process the communication between the storage systems fails, the process also fails. In addition, if communication fails after the migration reaches synchronized status, writes from the host will fail if the *source updating* option was chosen. The situation is further explained in 9.2, “Handling I/O requests” on page 248. The process of migrating data is performed at a volume level and as a background process.

The Data Migration Utility in XIV software revisions 10.2.2 and later supports these items:

- Up to eight migration targets can be configured on an XIV (where a target is either one controller in an active/passive storage device or one active/active storage device). The target definitions are used for both remote mirroring and data migration. Both remote mirroring and data migration functions can be active at the same time. An active/passive storage device with two controllers can use two target definitions provided that the migration volumes are balanced between both controllers.
- The XIV can communicate with host LUN IDs ranging from 0 to 511 (in decimal). This does not necessarily mean that the non XIV disk system can provide LUN IDs in that

range. You might be restricted by the ability of the non XIV storage controller to use only 16 or 256 LUN IDs depending on the hardware vendor and device.

- Up to 512 LUNs can be concurrently migrated, though this is not recommended.

Important: In this chapter, the source system in a data migration scenario is referred to as a *target* when setting up paths between the XIV Storage System and the donor storage (the non XIV storage). This is because the XIV is acting as a host/initiator and the source storage system is therefore the target. This terminology is also used in remote mirroring, and both functions share the same terminology for setting up paths for transferring data.

9.2 Handling I/O requests

The XIV Storage System handles all I/O requests for the host server during the data migration process. All read requests are handled based on where the data currently resides. For example, if the data has already been migrated to the XIV Storage System, it is read from that location. However, if the data has not yet been migrated to the IBM XIV storage, the read request comes from the host to the XIV Storage System, which in turn retrieves the data from the source storage device and provides it to the host server.

The XIV Storage System handles all host server write requests and the non XIV disk system is now transparent to the host. All write requests are handled using one of two user-selectable methods, chosen when defining the data migration. The two methods are known as *source updating* and *no source updating*.

An example of selecting which method to use is shown in Figure 9-2. The check box must be selected to enable source updating, shown here as Keep Source Updated. Without this check box selected, changed data from write operations is only written to the XIV.

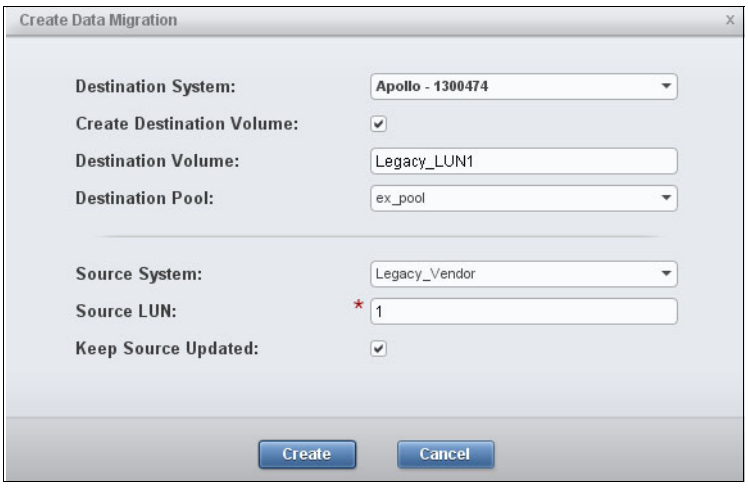


Figure 9-2 Keep Source Updated check box

Source updating

This method for handling write requests ensures that both storage systems (XIV and non XIV storage) are updated when a write I/O is issued to the LUN being migrated. By doing this the source system remains updated during the migration process, and the two storage systems remain in sync after the background copy process completes. Similar to synchronous Remote Mirroring, the write commands are only acknowledged by the XIV Storage System to the host

after writing the new data to the local XIV volume, then writing to the source storage device, and then receiving an acknowledgement from the non XIV storage system.

An important aspect of selecting this option is that if there is a communication failure between the target and the source storage systems or any other error that causes a write to fail to the source system, the XIV Storage System also fails the write operation to the host. By failing the update, the systems are guaranteed to remain consistent. Change-management requirements determine whether you choose to use this option.

Note: The advice is to use source updating. Updating the source storage system gives you the possibility to fall back to the source storage system in the event of a failure.

No source updating

This method for handling write requests ensures that only the XIV volume is updated when a write I/O is issued to the LUN being migrated. With the source updating off, write requests are only written to the XIV volume and are not written to the non XIV storage system. This is a preferred method for offline migrations. However, for online migrations where the server is using the XIV for read and write I/O, keeping this option off will limit your ability to back out of a migration. If the Keep Source Updated option is off during an online migration, you should have another way of recovering updates that were written to the volume being migrated after migration began. If the host is being shut down for the duration of the migration, or on and only used to read data, then this risk is mitigated.

Note: Do not select **Keep Source Updated** if you migrate a boot LUN. This is so you can quickly back out of a migration of the boot device if a failure occurs.

9.3 XIV and source storage connectivity

This section describes several considerations regarding the connectivity between the new XIV system and the source storage being migrated.

9.3.1 Multipathing with data migrations

Three types of storage systems exist where multipathing is concerned:

- Active/active: With these storage systems, volumes can be active on all of the storage system controllers at the same time (whether there are two controllers or more). These systems support I/O activity to any given volume down two or more paths. These types of systems typically support load balancing capabilities between the paths with path failover and recovery in the event of a path failure. The XIV is such a device and can utilize this technology during data migrations. Examples of IBM products that are active/active storage servers are the IBM DS6000, IBM DS8000, ESS F20, ESS 800, and SAN Volume Controller. The DS6000 and SAN Volume Controller are examples of storage servers that have preferred controllers on a LUN-by-LUN basis; if attached hosts ignore this preference, a potential consequence is the risk of a small performance penalty.

  If your non XIV disk system supports active/active then you can carefully configure multiple paths from XIV to non XIV disk. The XIV load balances the migration traffic across those paths and it automatically handles path failures.

- Active/passive: With these storage systems, any given volume can be active on only one controller at a time. These storage devices do *not* support I/O activity to any given volume down multiple paths at the same time. Most support active volumes on one or more

controllers at the same time, but any given volume can only be active on one controller at a time. Examples of IBM products that are active/passive storage devices are the DS4700 and the DCS3700.

- Asymmetric Logical Unit Access (ALUA): These storage systems are essentially active/passive multipathing systems with some intelligence built in. These system have a preferred path but can switch the owning controller depending on where I/O requests originate. Different implementations of ALUA exist, each with its own nuances. A strong recommendation is that ALUA be deactivated and connectivity between the XIV and source storage system be set up as active/passive. Because XIV is an active/active storage system, it will request I/O down all defined paths. This activity can lead to a "ping-pong" affect as the source storage system switches the LUN's owning controller back and forth from controller to controller. This, in turn, can lead to severe performance issues during the migration.

Migrating from an active/active storage device

If your non XIV storage system supports active/active LUN access then you can configure multiple paths from XIV to the non XIV disk system. The XIV load balances the migration traffic across these paths. This might tempt you to configure more than two connections or to increase the initialization speed to a large value to speed up the migration. However, the XIV can synchronize only one volume at a time per target (with four targets, this means that four volumes can be migrated at once). This means that the speed of the migration from each target is determined by the ability of the non XIV storage system to read from the LUN currently being migrated. Unless the non XIV storage system has striped the volume across multiple RAID arrays, the migration speed is unlikely to exceed 250-300 MBps (and could be much less), but this is totally dependant on the non XIV storage system.

If the other storage is going to be used by other servers while data is being migrated to the XIV, care must be taken to not overwork the other storage and cause latency on the servers that are not being migrated. It is best to leave the XIV migration speed set to the defaults and start migrating slowly to see how the existing environment can handle the change. As comfort levels rise and the migration process is learned, settings can be changed and more server LUNs can be moved.

Important: If multiple paths are created between an XIV and an active/active storage device, the same SCSI LUN IDs to Host IDs associations must be used for each LUN on each path, or data corruption will occur. Configure a maximum of two paths per target. Defining more paths will not increase throughput. With some storage arrays, defining more paths adds complexity and increases the likelihood of configuration issues and corruption.

Migrating from an active/passive storage device

Because of the active/active nature of XIV, special considerations must be made when migrating data from an active/passive storage device to XIV. A single path is configured between any given non XIV storage system controller and the XIV system. Many users decide to perform migrations with the host applications offline, because of the single path.

Define the target to the XIV for each non XIV storage controller (controller, not port). Define at least one path from that controller to the XIV. All volumes active on the controller can be migrated using the defined target for that controller. For example, suppose the non XIV storage system contains two controllers, A and B, (see Figure 9-3 on page 251):

- Define one target (called ctrl-A, for example) with one path between the XIV and one controller on the non XIV storage system (for example, controller *A*). All volumes active on this controller can be migrated by using this target. When defining the XIV initiator to the controller, be sure to define it as *not* supporting fail-over or ALUA if these options

are available on the non XIV storage array. By doing so, volumes that are passive on controller *A* are not presented to the XIV. To do this, see your non XIV storage system documentation.

- Define another target (called ctrl-B, for example) with one path between the XIV and controller *B*. All volumes active on controller *B* can be migrated to the XIV by using this target. When defining the XIV initiator to the controller, be sure to define it as *not* supporting failover or ALUA if these options are available. By doing so, volumes that are passive on controller *B* are not presented to the XIV. To do this, see your non XIV storage system documentation.

Figure 9-3 Active/Passive as multiple targets

Notes:

- If a single controller has two target ports (DS4700, for example), both can be defined as links for that controller target. Make sure that the two target links are connected to separate XIV modules for redundancy in case of a module failure. However, the migration will fail should the lun being migrated trespass (ownership transfers) to the other controller.
- Certain examples in this chapter are from an IBM DS4000® active/passive migration with each DS4000 controller defined independently as a target to the XIV Storage System. If you define a DS4000 controller as a target, do not define the alternate controller as a second port on the first target. Doing so causes unexpected issues such as migration failure, preferred path errors on the DS4000, slow migration progress, or corruption. See Figure 9-4 and Figure 9-5.

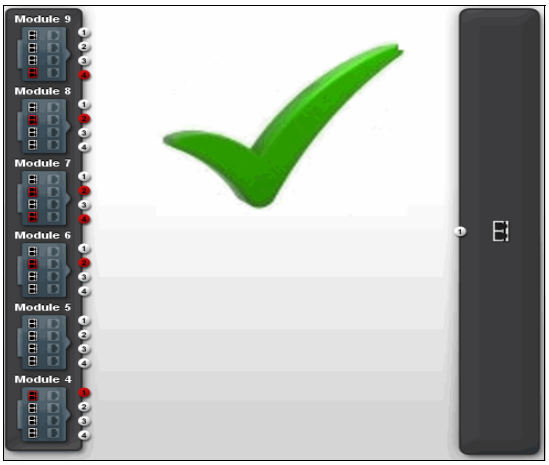


Figure 9-4 Active/Passive single port

Figure 9-5 Active/Passive dual port

9.3.2 Understanding initialization rate

The rate at which data is copied between the source storage system and the XIV is configurable. After a data migration is defined and activated, an XIV background process copies or pulls data from the source storage system to the XIV. The background process starts at sector 0 of the source LUN and copies the data sequentially to the end of the source LUN/volume. The rate at which the data is copied is called the *initialization rate*. The default maximum initialization rate is 100 MBps but is configurable. An important aspect to remember is that as XIV receives read requests from the server where the data has not yet been copied, the XIV must request the data from the source LUN, get the data, and then pass that data to the server. Also, note that Keep Source Updated option is a synchronous operation where the XIV must send the server's write request to the source storage system and waits for an acknowledgement before acknowledging the write to the server. These "real-time" requests are done outside of the background process and are therefore not limited by the maximum initialization rate. However, many mid-tier storage systems cannot maintain the default 100 MBps sync rate and the real-time read/writes. In these cases, the server seems to stall or hang on real-time reads and writes, especially on boot up. The real-time reads and writes will queue up behind the background copy reads if the LUN cannot maintain the initialization rate. To minimize the impact to the server, be sure to set the Max Initialization Rate to no more than 50 MBps for mid-tier storage systems; in many cases the suggested setting should be 30 MBps.

Also, be sure to understand the configuration of the source LUN being migrated because the RAID type and number of underlying disks or stripe width will have a large impact on how much data the LUN can supply. If, upon server boot, the server appears to stall or takes longer to boot than normal with actively migrating LUNs, decrease the Initialization Rate setting. Even large high-end storage systems are susceptible to misconfigured LUNs that can affect server response times when the Max Initialization Rate is set too high. Always be aware that increasing the sync rate too high can have a negative impact on the application. The idea of online data migrations is to *not* affect the applications. Whether a migration takes two hours or six does not matter, if data remains accessible and the server response meets SLAs.

Note: If a server appears to stall or take longer to boot than normal with actively migrating LUNs, decrease the Max Initialization Rate. The issue is that the source LUN cannot provide data fast enough. It cannot keep up with Initialization rate and the real-time read/write.

Testing has shown that setting a sync rate higher than what the source LUN or storage system can supply is counter-intuitive and increases the migration time because of low-level protocol aborts and retries.

9.4 Data migration steps

At a high level, the steps to migrate a volume from a non XIV system to the IBM XIV Storage System are as follows:

1. Initial connection and pre-implementation activities:
   - Cable and zone the XIV to the non XIV storage device.
   - Define XIV on the non XIV storage device (as a Linux or Windows host).
   - Define non XIV storage device on the XIV (as a migration target).
   - Update relevant OS patches, drivers, and HBA firmware on the host to the most current version that is supported by the non XIV storage system vendor.
2. Perform pre-migration tasks for each host that is being migrated:
   - Back up your data.
   - Stop all I/O from the host to the LUNs on the non XIV storage.
   - Shut down the host.
   - Remove non XIV multipath drivers; install IBM Host Attach Kit and any other required drivers or settings. Note, rebooting might required and is recommended.
3. Define and test the data migration volume:
   - On non XIV storage, remap volumes from the old host to the XIV host.
   - On XIV, create data migration tasks and test them.
4. Activate data migration tasks on XIV.
5. Define the host on XIV and bring the host and applications online:
   - Zone the host to XIV.
   - Define host and WWPN on XIV.
   - Map volumes to the host on XIV.
   - Bring the host online.
   - Verify XIV storage visible.
   - Enable/start host and applications.
6. Complete the data migration on XIV.
   - Monitor the XIV migration tasks.
   - On completion, delete the migration tasks.
7. Perform post migration activities.
   - In this case, remove zones between host and non XIV storage.

Tip: Print these overview steps and refer to them when you perform a migration.

Details of these steps are explained in the following sections.

9.4.1 Initial connection and pre-implementation activities

For the initial connection setup, start by cabling and zoning the XIV to the system being migrated.

If the migration or the host attachment (or both) is through iSCSI then you must ensure any firewalls are opened to allow iSCSI communications.

Important: If the IP network includes firewalls, TCP port 3260 must be open for iSCSI host attachment and replication to work.

It is also a good opportunity for the host OS patches, drivers, and HBA firmware to be updated to the latest supported levels for the non XIV storage.

Cable and zone the XIV to the non XIV storage system

Because the non XIV storage system views the XIV as a Linux host, the XIV must connect to the non XIV storage system as a SCSI initiator. Therefore, the physical connection from the XIV must be from initiator ports on the XIV (which by default is Fibre Channel port 4 on each active interface module). The initiator ports on the XIV must be fabric-attached (in which case they must be cabled through a fiber switch and then zoned to the non XIV storage system). Use two physical connections from two separate modules on two separate fabrics for redundancy if the non XIV storage system is active/active (redundant pathing is not possible on active/passive controllers).

The possibility exists that the host is attached through one protocol (such as iSCSI), and the migration occurs through the other (such as Fibre Channel). The host-to-XIV connection method and the data migration connection method are independent of each other.

Depending on the non XIV storage system vendor and device, an easier approach might be to zone the XIV to the ports where the volumes being migrated are already present. In this manner, the non XIV storage system might not require reconfiguration. For example, in EMC Symmetrix/DMX environments, zoning the fiber adapters (FAs) to the XIV where the volumes are already mapped might be an easier task.

When you are done, you will have completed these steps:

1. Run cables from XIV patch panel port 4 (initiator port) on each selected XIV interface module to a fabric switch.
2. Zoned the XIV initiator ports (whose WWPNs end in 3) to the selected non XIV storage system host ports using single initiator zoning. Each zone should contain one initiator port and one target port.

Figure 9-6 depicts a fabric-attached configuration. It shows that module 4 port 4 is zoned to a port on the non XIV storage through fabric A. Module 7 port 4 is zoned to a port on the non XIV storage through fabric B.

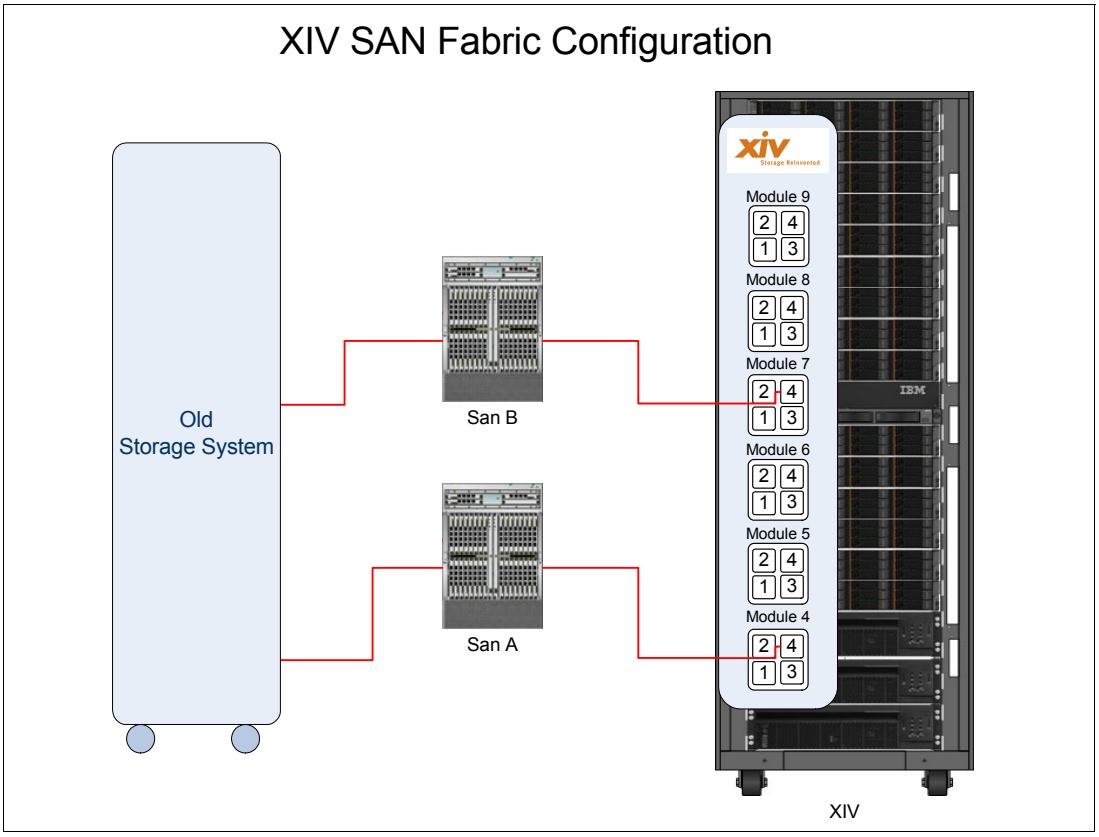


Figure 9-6 Fabric attached

Define XIV on the non XIV storage system (as a host)

After the physical connection and zoning between the XIV and non XIV storage system is complete, the XIV initiator (WWPN) must be defined on the non XIV storage system (remember, the XIV is nothing more than a Linux host to the non XIV storage system). The process to achieve this depends on vendor and device because you must use the non XIV storage system management interface. See the non XIV storage vendor's documentation for information about how to configure hosts to the non XIV storage system because the XIV is seen as a host to the non XIV storage.

If you have already zoned the XIV to the non XIV storage system, the WWPNs of the XIV initiator ports (which end in the number 3) are displayed in the WWPN drop-down list. This is dependent on the non XIV storage system and storage management software. If they are not there, you must manually add them (this might imply that the SAN zoning has not been done correctly).

The XIV must be defined as a Linux or Windows host to the non XIV storage system. If the non XIV system offers several variants of Linux, you can choose SUSE Linux, Red Hat Linux, or Linux x86. This defines the correct SCSI protocol flags for communication between the XIV and non XIV storage system. The principal criterion is that the host type must start LUN numbering with LUN ID 0. If the non XIV storage system is active/passive, determine whether the host type selected affects LUN failover between controllers, such as DS4000 (see 9.14.5, "IBM DS3000, DS4000, DS5000" on page 308 for more details).

Other vendor-dependent settings might also exist. See 9.14, “Device-specific considerations” on page 304 for additional information.

Define non XIV storage system on the XIV (as a migration target)

After the physical connectivity is made and the XIV is defined to the non XIV storage system, the non XIV storage system must be defined on the XIV. This includes defining the storage system object, defining the WWPN ports on the non XIV storage system, and defining the connectivity between the XIV and the non XIV storage system. Complete the following steps:

1. In the XIV GUI go to the **Remote** → **Migration Connectivity** panel.
2. Click **Create Target** (Figure 9-7).

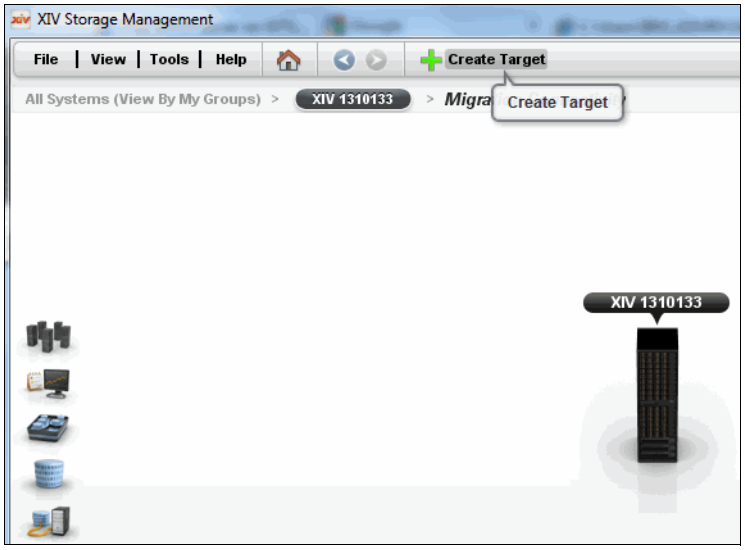


Figure 9-7 Create target for the non XIV storage device

Note: If Create Target is disabled and cannot be clicked, you have reached the maximum number of targets; targets are both migration targets and mirror targets.

3. The window shown in Figure 9-8 on page 257 opens. Make the appropriate entries and selections, and then click **Create**.
   - Target Name: Enter a name of your choice.
   - Target Protocol: Select **FC** from the pull-down menu.
   - Max Initialization Rate: This is the rate at which the XIV background copy tries to pull or copy data from the source, non XIV storage system. When migrating from a mid-tier storage system, the Max Initialization Rate should not exceed 50 MBps and in many cases should not exceed 30. Most mid-tier storage systems cannot maintain the default 100 MBps background copy rate for a single LUN and maintain acceptable real-time reads/writes (this is where the server requests data that has not yet been copied and the XIV must get the data from the source storage system and pass on to the host).

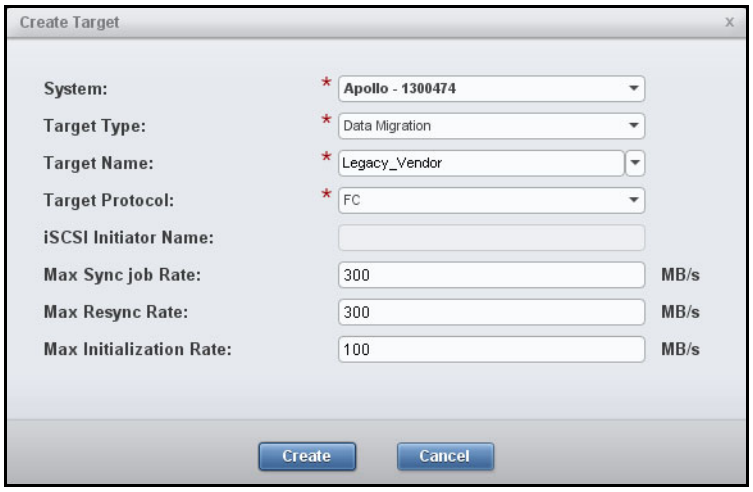


Figure 9-8 Defining the non XIV storage device

Tip: The data migration target is represented by an image of a generic rack. If you must delete or rename the migration device, do so by right-clicking the image of that rack.

If this is the first time you have used data migration and the XIV GUI tips are on, a Tip window opens and gives you more information (Figure 9-9).

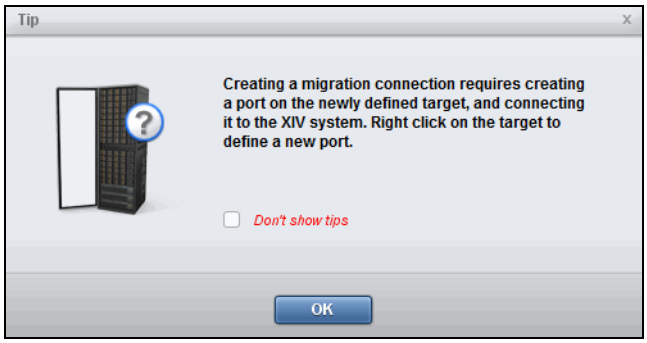

Tip
Creating a migration connection requires creating a port on the newly defined target, and connecting it to the XIV system. Right click on the target to define a new port.
Don't show tips
OK

Figure 9-9 Data migration Tip

4. Right-click the dark box that is part of the defined target and click **Add Port** (Figure 9-10 on page 258).

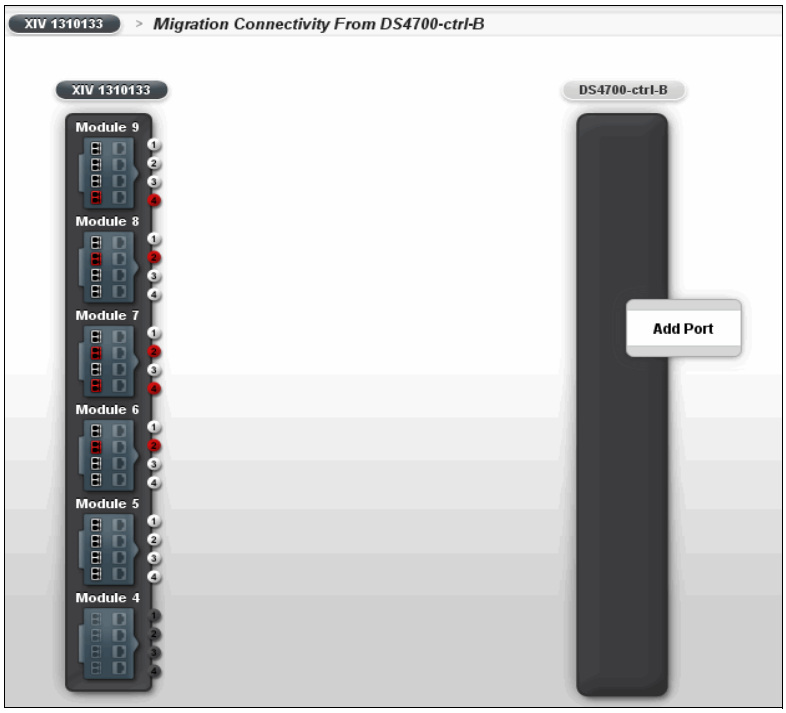


Figure 9-10 Defining the target port

The Add Port dialog opens. Do the following steps:

a. Enter the WWPN of the first (fabric A) port on the non XIV storage system zoned to the XIV. There is no drop-down menu of WWPNs, so you must manually type or paste in the correct WWPN. Be careful not to make a mistake. Using colons to separate every second number is unnecessary. For example, you can use either of the following formats to enter a number for the WWPN; it makes no difference which format you use:

- `10:00:00:c9:12:34:56:78`
- `100000c912345678`

b. Click **Add**.

5. Enter another port (repeating step 4 on page 257) for those storage devices that support active/active multipathing. This can be the WWPN that is zoned to the XIV on a separate fabric.
6. Connect the XIV and non XIV storage ports that are zoned to one another. This is done by clicking and dragging from port 4 on the XIV to the port (WWPN) on the non XIV storage system to which the XIV is zoned. In Figure 9-11 on page 259 the mouse started at module 8 port 4 and has nearly reached the target port. The connection is currently colored blue; it changes to red when the line reaches port 1 on the target.

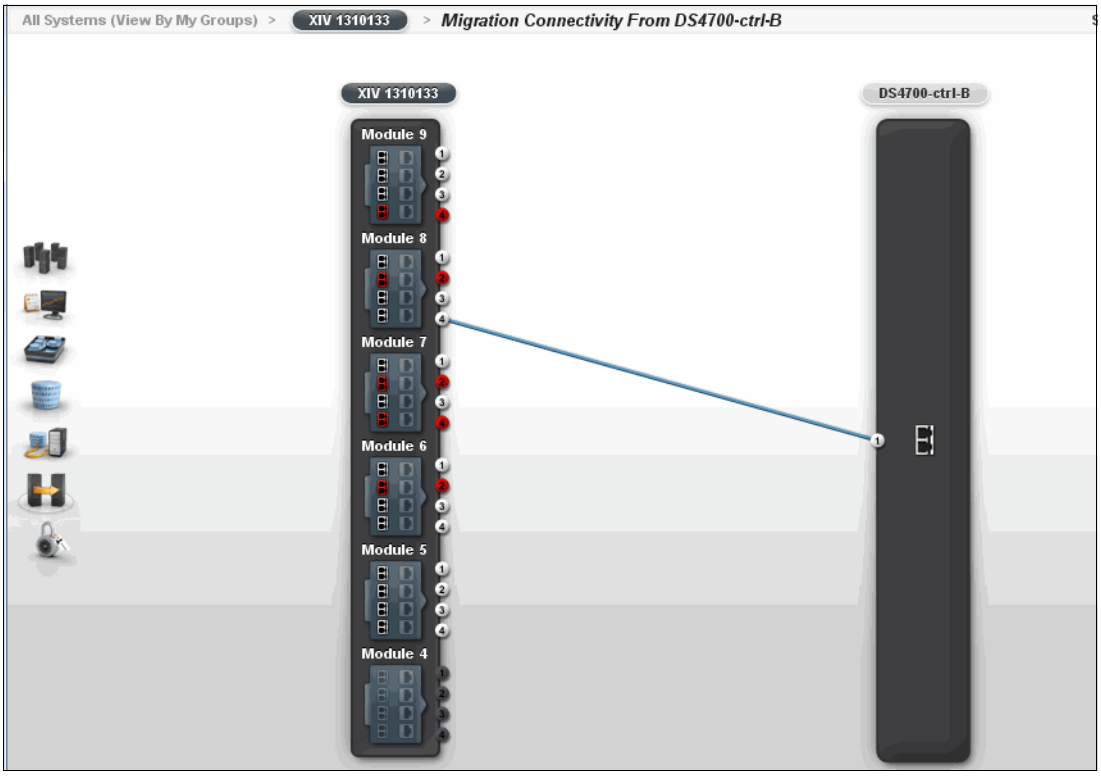


Figure 9-11 Dragging a connection between XIV and migration target

In Figure 9-12 on page 260, the connection from module 8 port 4 to port 1 on the non XIV storage system is currently active, as noted by the green color of the connecting line. This means that the non XIV storage system and XIV are connected and communicating. (This indicates that SAN zoning was done correctly. The correct XIV initiator port was selected. The correct target WWPN was entered and selected, and LUN 0 was detected on the target device). If there is an issue with the path, the connection line is red.

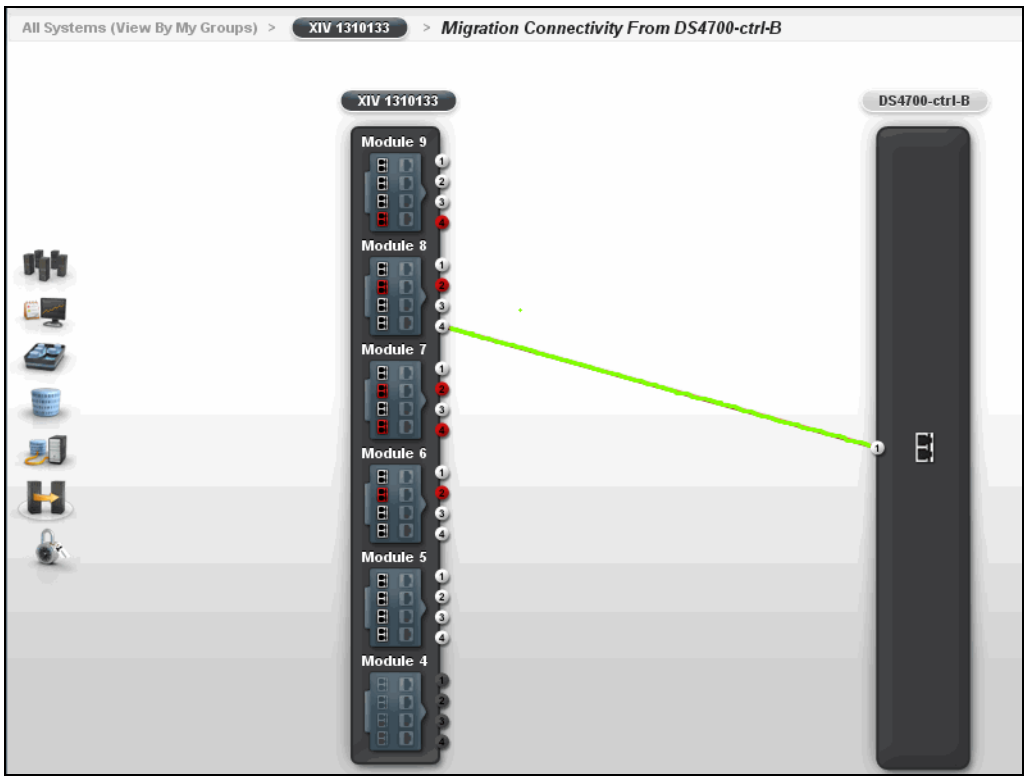


Figure 9-12 non XIV storage system defined

Tip: Depending on the storage controller, ensuring that LUN0 is visible on the non XIV storage system down the controller path that you are defining helps ensure proper connectivity between the non XIV storage system and the XIV. Connections from XIV to DS4000, EMC DMX, or Hitachi HDS devices require a real disk device to be mapped as LUN0. However, the IBM Enterprise Storage Server, ESS 800, for instance, does not need a LUN to be allocated to the XIV for the connection to become active (turn green in the GUI). The same is true for the EMC CLARiiON.

Update OS patches, drivers, and HBA firmware for non XIV storage

Before proceeding to migrate, be sure that the host has all its OS patches, drivers, and HBA firmware are at the latest levels supported by the non XIV storage. This is important so the host can support attachment to either storage array in case there are reasons to roll back to the old environment.

If XIV attachment requires newer versions of the patches, driver, and firmware that is supported by non XIV storage, an extra step is required during the installation of the XIV Host Attachment Kit, as detailed in “Enable or Start host and applications” on page 268.

Check the following common components for newer levels:

- OS patches (for example, hotfixes)
- HBA firmware
- HBA drivers
- Multipathing drivers (including MPIO DSMs)

See the non XIV storage support site to see which specific component levels are supported and whether any guidance on levels for related components (like OS patches) is provided.

Then check supported levels on the XIV support pages to ensure you are running the latest component levels that are commonly supported.

9.4.2 Perform pre-migration tasks for each host being migrated

Perform the following pre-migration tasks just before the LUNs are redirected to the XIV and migration tasks are defined.

Back up the volumes being migrated

A full restorable backup must be created prior to any data migration activity. The best practice is to verify the backup, that all the data is restorable, and that there are no backup media errors.

In addition to a regular backup, a point-in-time copy of the LUNs being migrated (if available) is an extra level of protection allowing you to perform a rapid rollback.

Stop all I/O from the host to the LUNs on the non XIV storage

Before the actual migration can begin the application must be quiesced and the file system synchronized. This ensures that the application data is in a consistent state. Because the host might need to be rebooted several times prior to the application data being available again, the following steps might be required:

- Set applications to not automatically start when the host operating system restarts.
- Stop file systems from being automatically remounted on boot. For UNIX-based operating systems consider commenting out all affected file system mount points in the fstab or vfstab.

Note: In clustered environments like Windows or Linux, you could choose to work with only one node until the migration is complete; if so, consider shutting down all other nodes in the cluster.

Remove non XIV multipath driver and install XIV Host Attachment Kit

The XIV Host Attachment Kit should be installed on all platforms that it is available for, even if you are using a supported alternative multipathing driver.

Before the XIV Host Attach is installed, remove the old multipathing driver. Make sure you have backup copies of any of the non XIV software, drivers, and so on, in case you need to fall back to an earlier configuration.

Update any additional HBA firmware, drivers, and patches that have been identified as a requirement for XIV attachment.

Important: If the LUN being migrated is a boot volume, the existing HBA hardware, firmware, and drivers must be upgraded to a level that supports the XIV.

Check the IBM support site, *XIV Host Attachment Guide* and *Release Notes* for the latest versions of the HBA firmware, drivers, and patches. Download and install the software as appropriate.

The XIV documentation and software can be accessed from the IBM support portal:

`http://www.ibm.com/support/entry/portal/Downloads`

9.4.3 Define and test data migration volume

Best practice is to always test before you try anything the first time. This allows for time to learn the process and gain confidence. Follow these steps to define and test the data migration volume:

1. Map a non XIV volume to XIV.

   The volumes being migrated to the XIV must be allocated through LUN mapping to the XIV. The LUN ID presented to the XIV must be a decimal value in the range of 0 - 511. If it uses hexadecimal LUN numbers then the LUN IDs can range from 0x0 to 0x1FF, but must be converted to decimal when entered into the XIV GUI. The XIV does not recognize a host LUN ID above 511 (decimal). Figure 9-13 shows LUN mapping using a DS4700. It depicts the XIV as a host called XIV_Migration_Host with four DS4700 logical drives mapped to the XIV as LUN IDs 0 to 3.

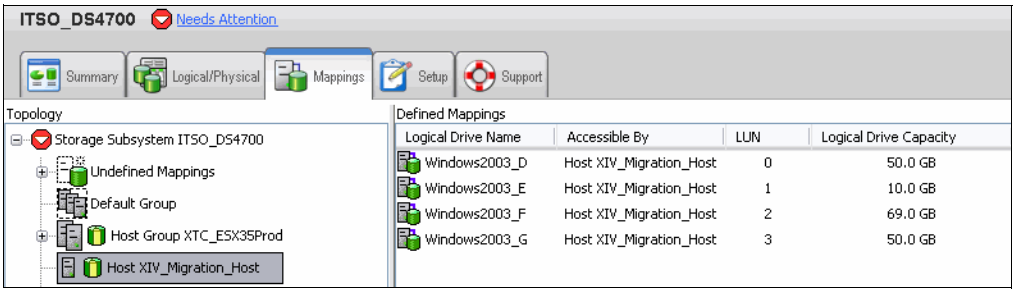


Figure 9-13 Non XIV LUNs defined to XIV

When mapping volumes to the XIV, note the SCSI LUN IDs/Host IDs allocated by the non XIV storage. The methodology to do this varies by vendor and device and is documented in greater detail in 9.14, “Device-specific considerations” on page 304.

Important: You must unmap the volumes away from the host during this step, even if you plan to power the host off during the migration. The non XIV storage only presents the migration LUNs to the XIV. Do not allow a possibility for the host to detect the LUNs from both the XIV and the non XIV storage.

2. Define data migration object/volume.

   After the volume being migrated to the XIV is mapped to the XIV, a new data migration (DM) volume can be defined. The source volume from the non XIV storage system and the XIV volume must be exactly the same size; therefore, in most cases it is easiest to let XIV create the XIV LUN for you, as discussed in the following section.

Important: You cannot use the XIV data migration function to migrate data to a source volume in an XIV remote mirror pair. If you need to do this, migrate the data first and then create the remote mirror after the migration is completed.

If you want to manually pre-create the volumes on the XIV, go to 9.6, “Manually creating the migration volume” on page 274. However, the preferred way is to automatically create, as described next.

XIV volume automatically created

The XIV has the ability to determine the size of the non XIV volume and create the XIV volume quickly when the data migration object is defined. Use this method to help avoid potential problems when manually calculating the real block size of a volume:

1. In the XIV GUI, go to the function menu **Remote** → **Migration** (Figure 9-14).

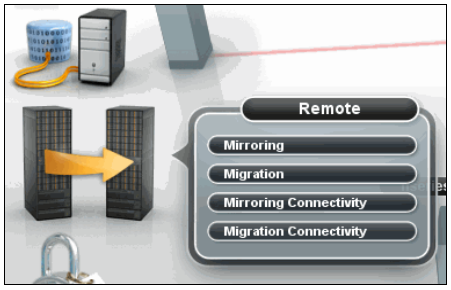


Figure 9-14 Remote function menu

2. Click the **Create Data Migration** option from the menu bar.
3. The Create Data Migration window opens (Figure 9-15).

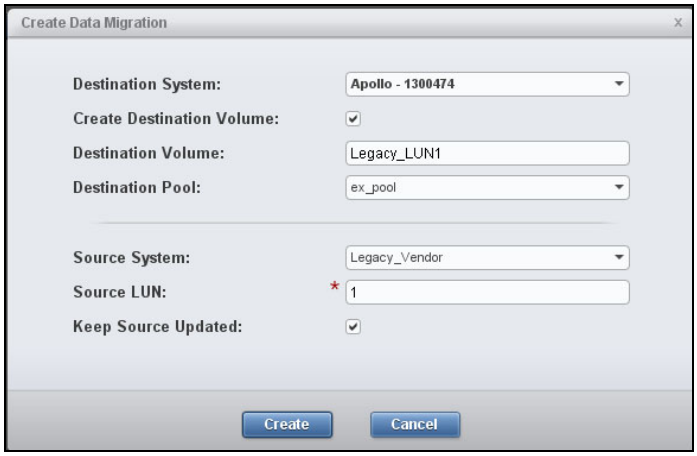


Figure 9-15 Define Data Migration object/volume

Make the appropriate selections and entries:

- Destination Pool: Choose the pool from the drop-down menu where the volume will be created.
- Destination Name: Enter a user-defined name. This will be the name of the local XIV volume.
- Source Target System: Choose the already defined non XIV storage system from the drop-down menu.

> **Important:** If the non XIV source device is active/passive, then the source target system must represent the controller (or service processor) that currently owns the source LUN being migrated. This means that you must check, from the non XIV storage, which controller is presenting the LUN to the XIV.

– Source LUN: Enter the decimal value of the host LUN ID as presented to the XIV from the non XIV storage system. Certain storage devices present the LUN ID as hex. The number in this field must be the decimal equivalent. Ensure that you do not accidentally use internal identifiers (IDs) that you might also see on the source storage systems management panels. In Figure 9-13 on page 262, the correct values to use are in the LUN column (numbered 0 to 3).

Note: Confusing Internal LUN IDs or Names to SCSI LUN IDs/Host IDs is the most often seen mistake. When creating DMs on the XIV, be sure to use the SCSI LUN IDs/Host ID in the Source LUN. This is the SCSI ID/Host ID as presented to the XIV.

– Keep Source Updated: Select this if the non XIV storage system source volume is to be updated with writes from the host. In this manner all writes from the host will be written to the XIV volume, and also the non XIV source volume, until the data migration object is deleted.

If migrating the boot LUN: Do *not* select Keep Source Updated if migrating the boot LUN. This way, you can quickly back out of a migration of the boot device if a failure occurs.

4. Click **Define**. The migration is displayed, as shown in Figure 9-16.

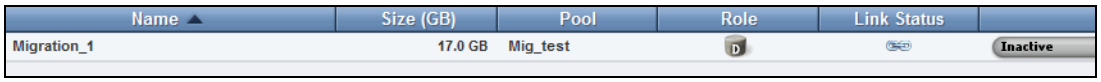


Figure 9-16 Defined data migration object/volume

Note: Define Data Migration will query the configuration of the non XIV storage system and create an equal sized volume on XIV. To check if you can read from the non XIV source volume you need to Test Data Migration. On some active/passive non XIV storage systems the configuration can be read over the passive controller, but Test Data Migration will fail.

5. The only way to show the pool in which the migration volume was created is if the `Pool` column is being displayed. If the Pool column is missing, right-click any of the column titles and the Customize Columns dialog box will be displayed (Figure 9-17 on page 265). Select **Pool** from the Hidden Columns list and click the right arrow to add Pool to the Visible Columns list. Figure 9-18 on page 265 shows the Pool column information for the LUN names. Column placement cannot be changed.

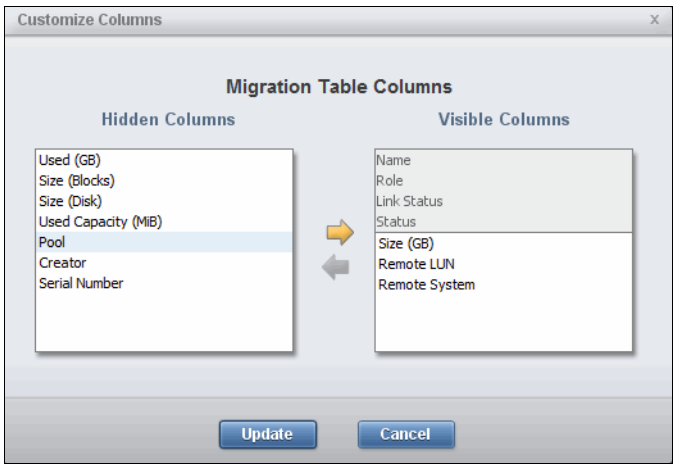


Figure 9-17 Customize Columns dialog box

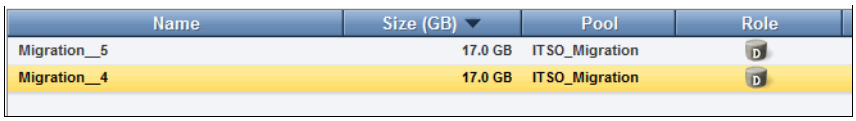


Figure 9-18 Pool column

6. Test the data migration object. Right-click to select the created data migration object and choose **Test**. If there are any issues with the data migration object the test fails and the issues encountered are reported (Figure 9-19).

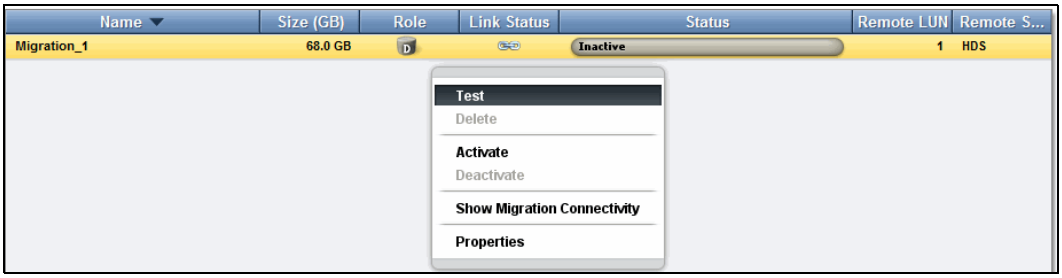


Figure 9-19 Test data migration

Tip: If you are migrating volumes from a Microsoft Cluster Server (MSCS) that is still active, testing the migration might fail because of the reservations placed on the source LUN by MSCS. You must bring the cluster down properly to get the test to succeed. If the cluster is not brought down properly, errors will occur either during the test or when activated. The SCSI reservation must then be cleared from the source storage system for the migration to succeed. Review source storage system documentation for how to clear SCSI reservations.

9.4.4 Activate a data migration on XIV

After the data migration volume has been tested the process of the actual data migration can begin. When data migration is initiated, the data is copied sequentially in the background from the non XIV storage system volume to the XIV. The host reads and writes data to the XIV storage system without being aware of the background I/O being performed.

Note: After it is activated, the data migration can be deactivated but is *not* recommended. When the data migration is deactivated, the host is no longer able to read or write to the source migration volume and all host I/O stops. Do *not* deactivate the migration with host I/O running. If you want to abandon the data migration prior to completion, use the backing out process described in 9.12, “Backing out of a data migration” on page 300.

Activate the data migration

Right-click to select the data migration object/volume and choose **Activate**. This begins the data migration process where data is copied in the background from the non XIV storage system to the XIV. Activate all volumes being migrated so that they can be accessed by the host. The host has read and write access to all volumes, but the background copy occurs serially volume by volume. If two targets (such as non XIV 1 and non XIV 2) are defined with four volumes each, two volumes are actively copied in the background: one volume from non XIV 1 and another from non XIV 2. All eight volumes are accessible by the hosts.

Figure 9-20 shows the menu choices when right-clicking the data migration. The Test Data Migration, Delete Data Migration, and Activate menu items are the most-used commands.

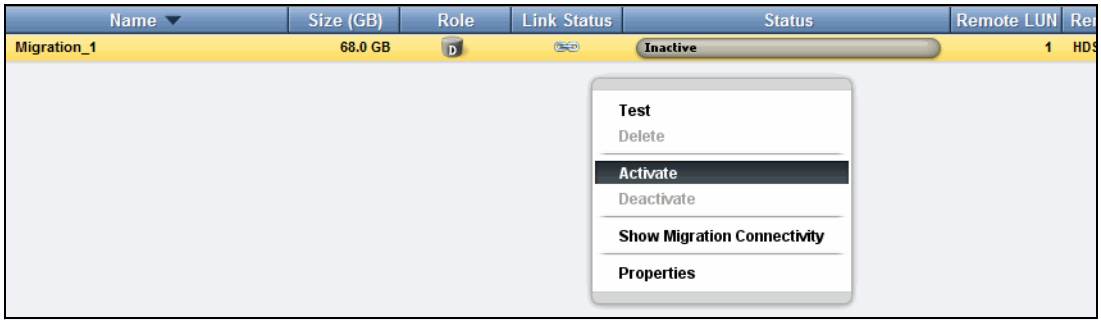


Figure 9-20 Activate data migration

9.4.5 Define the host on XIV and bring host online

The host must be directed (through SAN fabric zoning) to the XIV instead of the non XIV storage system, by using the following procedures:

- Disable the zone between the host and the non XIV storage.
- Enable the zone between the host and the XIV.

The XIV is acting as a proxy between the host and the non XIV storage system. The host must no longer access the non XIV storage system after the data migration is activated. The host must perform all I/O through the XIV.

Defining a zone between the host and XIV can be done prior to the migration, but you might have to disable the non XIV zone between the host and the non XIV storage at this point. This is because some storage systems might present a “LUN 0” for in-band management communications, causing issues after the host is brought back online.

For SAN boot volumes, define a zone with a single path from the server to the XIV. The migration should be run with the old multipath software installed and not removed until the migration is complete. After the data is confirmed complete then the other multipath drivers can be removed and MPIO can be properly configured for the XIV. This will help ensure that the server can go to back to the old storage if there is a problem.

Define the host being migrated to the XIV

Prior to performing data migrations and allocating the volumes to the host, the host must be defined on the XIV. Volumes are then mapped to the hosts or clusters. If the host is to be a member of a cluster, then the cluster must be defined first. However, a host can be moved easily from or added to a cluster at any time. This also requires that the host be zoned to the XIV target ports through the SAN fabric.

1. Optional: Define a cluster:
   a. In the XIV GUI go to the floating menu **Host and Clusters** → **Host and Clusters**.
   b. Choose **Add Cluster** from the top menu bar.
   c. Name: Enter a cluster name in the provided space.
   d. Click **OK**.
2. Define a host:
   a. In the XIV GUI go to the floating menu **Host and Clusters** → **Host and Clusters**.
   b. Choose **Add Host** from the top menu bar. Make the appropriate entries and selections.
      i. Name: Enter a host name.
      ii. Cluster: If the host is part of a cluster, choose the cluster from the drop-down list.
      iii. Click **Add**.
   c. Select the host and right-click to access the menu from which to choose **Add Port**.
      i. Port Type: Choose **FC** from the drop-down list.
      ii. Port Name: This produces a drop-down list of WWPNs that are logged into the XIV but that have not been assigned to a host. WWPNs can be chosen from the list or entered manually.
      iii. Click **Add**.
   d. Repeat these steps to add all the HBAs of the host being defined.

Map volumes to the host on XIV

After the data migration has been started, you can use the XIV GUI or XCLI to map the migration volumes to the host. When mapping volumes to hosts on the XIV, LUN ID 0 is reserved for XIV in-band communication. This means that the first LUN ID that you normally use is LUN ID 1. This includes boot-from-SAN hosts. You might also choose to use the same LUN IDs as were used on the non XIV storage, but this is not mandatory.

Important: The host cannot read the data on the non XIV volume until the data migration has been activated. The XIV does not pass through (proxy) I/O for a migration that is inactive. If you use the XCLI **`dm_list`** command to display the migrations, ensure that the word Yes is displayed in the Active column for every migration.

Enable or Start host and applications

The host can be started and LUNs checked that they are visible and usable.

When the volumes and data access have been confirmed, the application can be brought up and operations verified. The migration tasks will run in the background and allow normal host access to the newly mapped XIV volumes. If the application start procedures were modified according to 9.4.2, “Perform pre-migration tasks for each host being migrated” on page 261, the application startup procedures can now be configured to start as normal.

Restart the server a final time and confirm that all the drive letters are correct and that the applications started.

Occasionally, a host might not need to be online for the duration of the migration (such as after hours, not in production, or expected migration completed within the customer change window); it can remain offline and be bought back online after the migration is complete.

Note: In clustered environments, the usual recommendation is for only one node of the cluster be initially brought online after the migration is started, and that all other nodes be offline until the migration is complete. After complete, update all other nodes (driver, host attachment package, and so on), in the same way the primary node was during the initial outage.

9.4.6 Complete the data migration on XIV

To complete the data migration, perform the steps described in this section.

Data migration progress

Figure 9-21 shows the progress of the data migrations. The status bar can be toggled between GB remaining, percent complete, and hours/minutes remaining. Figure 9-21 shows two data migrations, one of which has started background copy and one that has not. Only one migration is being copied at this time because there is only one target.

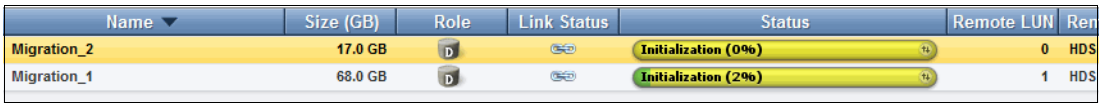


Figure 9-21 Data migration progress

After all of a volume’s data has been copied, the data migration achieves synchronization status. After synchronization is achieved, all read requests are served by the XIV Storage System. If source updating was selected the XIV will continue to write data to both itself and the outgoing storage system until the data migration is deleted. Figure 9-22 shows a completed migration.

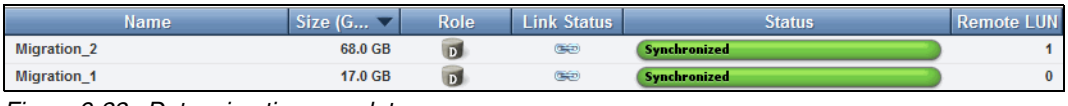


Figure 9-22 Data migration complete

Delete data migration

After the synchronization has been achieved, the data migration object can be safely deleted without host interruption.

Important: If this is an online migration, do *not* deactivate the data migration prior to *deletion*, as this causes host I/O to stop and possibly causes data corruption.

Right-click to select the data migration volume and choose **Delete Data Migration**, as shown in Figure 9-23. This can be done without host/server interruption.

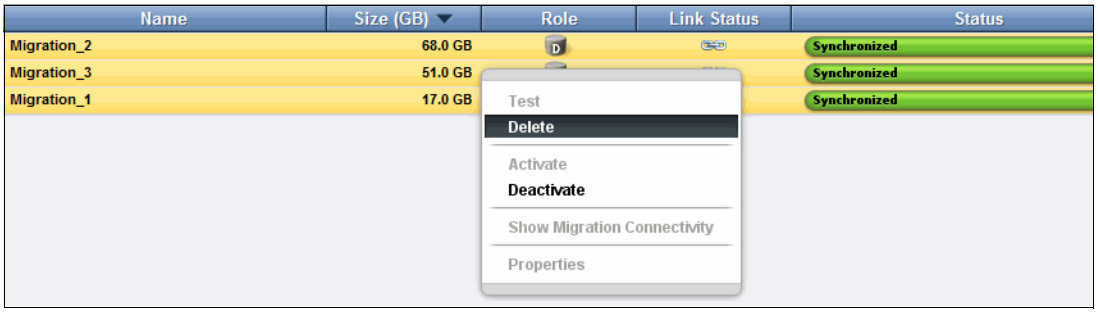


Figure 9-23 Delete data migration

Note: For safety purposes, you cannot delete an inactive or unsynchronized data migration from the Data Migration panel. An unfinished data migration can only be deleted by deleting the relevant volume from the Volumes → Volumes & Snapshots section in the XIV GUI.

9.4.7 Post migration activities

Typically these are cleanup activities performed after the migration has been completed, such as removing old zones between the host and non XIV storage and LUN mappings from the non XIV storage to XIV.

9.5 Command-line interface

All of the XIV GUI operation steps can be performed using the XIV command-line interface (XCLI) either through direct command execution or through batch files containing numerous commands. This is especially helpful in migration scenarios involving numerous LUNs. This section lists the XCLI command equivalent of the GUI steps shown previously. A full description of all the XCLI commands is in the *XIV Commands Reference* available at the following IBM website (select **IBM XIV Gen3** → **Publications**, and then click a specific version):

`http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp`

Every command issued in the XIV GUI is logged in a text file with the correct syntax. This is helpful for creating scripts. If you are running the XIV GUI under Microsoft Windows, search for a file named `guicommands_< todays date >.txt`.

To access this information from the GUI, click **Tools** and select **Commands Log**. Figure 9-24 on page 270 shows the Commands Log dialog box. From here the commands can be saved or cleared, or the commands collection process can be paused.

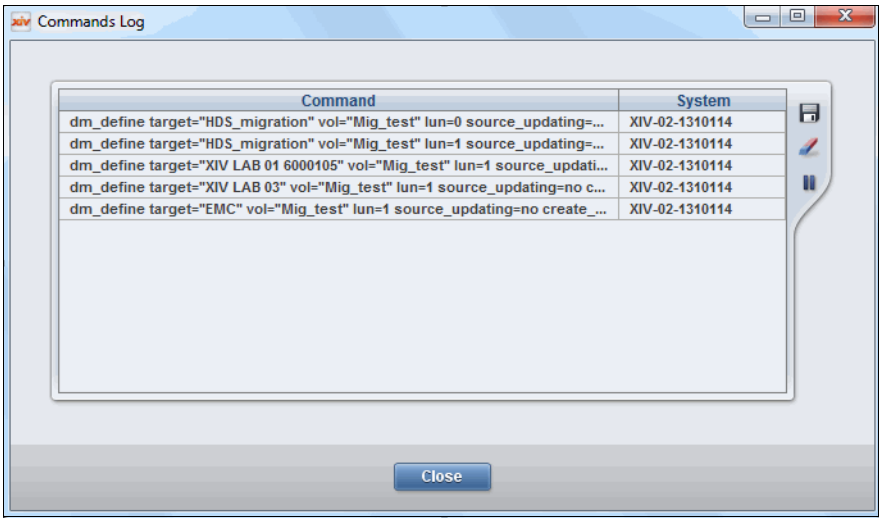


Figure 9-24 Commands Log

The following commands are effectively in the order in which you must execute them, starting with the commands to list all current definitions (which are also needed when you start to delete migrations).

- List targets:

  Syntax **`target_list`**

- List target ports:

  Syntax **`target_port_list`**

- List target connectivity:

  Syntax **`target_connectivity_list`**

- List clusters:

  Syntax **`cluster_list`**

- List hosts:

  Syntax **`host_list`**

- List volumes:

  Syntax **`vol_list`**

- List data migrations:

  Syntax **`dm_list`**

- Define target (Fibre Channel only):

  Syntax **`target_define target=<Name> protocol=FC xiv_features=no`**

  Example **`target_define target=DMX605 protocol=FC xiv_features=no`**

- Define target port (Fibre Channel only):

  Syntax **`target_port_add fcaddress=<non XIV storage WWPN> target=<Name>`**

  Example **`target_port_add fcaddress=0123456789012345 target=DMX605`**

- Define target connectivity (Fibre Channel only):

  Syntax **target_connectivity_define local_port=1:FC_Port:<Module:Port> fcaddress=<non XIV storage WWPN> target=<Name>**

  Example **target_connectivity_define local_port=1:FC_Port:5:4 fcaddress=0123456789012345 target=DMX605**

- Define cluster (optional):

  Syntax **cluster_create cluster=<Name>**

  Example **cluster_create cluster=Exch01**

- Define host (if adding host to a cluster):

  Syntax **host_define host=<Host Name> cluster=<Cluster Name>**

  Example **host_define host=Exch01N1 cluster=Exch01**

- Define host (if not using cluster definition):

  Syntax **host_define host=<Name>**

  Example **host_define host=Exch01**

- Define host port (Fibre Channel host bus adapter port):

  Syntax **host_add_port host=<Host Name> fcaddress=<HBA WWPN>**

  Example **host_add_port host=Exch01 fcaddress=123456789abcdef1**

- Create XIV volume using decimal GB volume size:

  Syntax **vol_create vol=<Vol name> size=<Size> pool=<Pool Name>**

  Example **vol_create vol=Exch01_sg01_db size=17 pool=Exchange**

- Create XIV volume using 512-byte blocks:

  Syntax **vol_create vol=<Vol name> size_blocks=<Size in blocks> pool=<Pool Name>**

  Example **vol_create vol=Exch01_sg01_db size_blocks=32768 pool=Exchange**

- Define data migration:

  If you want the local volume to be automatically created:

  Syntax **dm_define target=<Target> vol=<Volume Name> lun=<Host LUN ID as presented to XIV> source_updating=<yes|no> create_vol=yes pool=<XIV Pool Name>**

  Example **dm_define target=DMX605 vol=Exch01_sg01_db lun=5 source_updating=no create_vol=yes pool=Exchange**

  If the local volume was pre-created:

  Syntax **dm_define target=<Target> vol=<Pre-created Volume Name> lun=<Host LUN ID as presented to XIV> source_updating=<yes|no>**

  Example **dm_define target=DMX605 vol=Exch01_sg01_db lun=5 source_updating=no**

- Test data migration object:

  Syntax **dm_test vol=<DM Name>**

  Example **dm_test vol=Exch_sg01_db**

- Activate data migration object:

  Syntax **dm_activate vol=<DM Name>**

  Example **dm_activate vol=Exch_sg01_db**

- Map volume to host:

  Syntax **map_vol host=<Host Name> vol=<Vol Name> lun=<LUN ID>**

  Example **map_vol host=Exch01 vol=Exch01_sg01_db lun=1**

- Map volume to cluster:

  Syntax **map_vol host=<Cluster Name> vol=<Vol Name> lun=<LUN ID>**

  Example **map_vol host=Exch01 vol=Exch01_sg01_db lun=1**

- Delete data migration object:

  If the data migration is synchronized and thus completed:

  Syntax **dm_delete vol=<DM Volume name>**

  Example **dm_delete vol=Exch01_sg01_db**

  If the data migration is not complete, it must be deleted by removing the corresponding volume from the Volume and Snapshot menu (or through the **vol_delete** command).

- Delete volume (not normally needed):

  For a challenged volume delete (cannot be done through a script, because this command must be acknowledged):

  Syntax **vol_delete vol=<Vol Name>**

  Example **vol_delete vol=Exch_sg01_db**

  If you want to do an unchallenged volume deletion:

  Syntax **vol_delete -y vol=<Vol Name>**

  Example **vol_delete -y vol=Exch_sg01_db**

- Delete target connectivity:

  Syntax **target_connectivity_delete local_port=1:FC_Port:<Module:Port> fcaddress=<non XIV storage device WWPN> target=<Name>**

  Example **target_connectivity_delete local_port=1:FC_Port:5:4 fcaddress=0123456789012345 target=DMX605**

- Delete target port:

  Fibre Channel

  Syntax **target_port_delete fcaddress=<non XIV WWPN> target=<Name>**

  Example **target_port_delete fcaddress=0123456789012345 target=DMX605**

- Delete target:

  Syntax **target_delete target=<Target Name>**

  Example **target_delete target=DMX605**

- Change Migration Sync Rate:

  Syntax **target_config_sync_rates target=<Target Name> max_initialization_rate=<Rate in MB>**

  Example **target_config_sync_rates target=DMX605 max_initialization_rate=100**

9.5.1 Using XCLI scripts or batch files

To execute a XCLI batch job, the best approach is to use the XCLI (rather than the XCLI session).

Setting environment variables in Windows

You can remove the need to specify user and password information for every command by making that information an environment variable. Example 9-1 shows how this is done using a Windows command prompt. First the XIV_XCLIUSER variable is set to *admin*, then the XIV_XCLIPASSWORD is set to *adminadmin*. Then both variables are confirmed as set. If necessary, change the user ID and password to suit your setup.

Example 9-1 Setting environment variables in Microsoft Windows

```
C:\>set XIV_XCLIUSER=admin
C:\>set XIV_XCLIPASSWORD=adminadmin
C:\>set | find "XIV"
XIV_XCLIPASSWORD=adminadmin
XIV_XCLIUSER=admin
```

To make these changes permanent. complete the following steps:

1. Right-click the My Computer icon and select **Properties**.
2. Click the **Advanced** tab.
3. Click **Environment Variables**.
4. Click **New** for a new system variable.
5. Create the XIV_XCLIUSER variable with the relevant user name.
6. Click **New** again to create the XIV_XCLIPASSWORD variable with the relevant password.

Setting environment variables in UNIX

If your are using a UNIX-based operating system export the environment variables as shown in Example 9-2 (which in this example is AIX). In this example the user and password variables are set to *admin* and *adminadmin* and then confirmed as being set.

Example 9-2 Setting environment variables in UNIX

```
root@dolly:/tmp/XIVGUI# export XIV_XCLIUSER=admin
root@dolly:/tmp/XIVGUI# export XIV_XCLIPASSWORD=adminadmin
root@dolly:/tmp/XIVGUI# env | grep XIV
XIV_XCLIPASSWORD=adminadmin
XIV_XCLIUSER=admin
```

To make these changes permanent, update the relevant profile, being sure that you export the variables to make them environment variables.

Note: It is also possible to run XCLI commands without setting environment variables with the **-u** and **-p** switches.

9.5.2 Sample scripts

With the environment variables set, a script or batch file like the one in Example 9-3 can be run from the shell or command prompt to define the data migration pairings.

Example 9-3 Data migration definition batch file

```
xcli -m 10.10.0.10 dm_define vol=MigVol_1 target=DS4200_CTRL_A lun=4
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_2 target=DS4200_CTRL_A lun=5
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_3 target=DS4200_CTRL_A lun=7
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_4 target=DS4200_CTRL_A lun=9
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_5 target=DS4200_CTRL_A lun=11
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_6 target=DS4200_CTRL_A lun=13
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_7 target=DS4200_CTRL_A lun=15
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_8 target=DS4200_CTRL_A lun=17
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_9 target=DS4200_CTRL_A lun=19
source_updating=no create_vol=yes pool=test_pool
xcli -m 10.10.0.10 dm_define vol=MigVol_10 target=DS4200_CTRL_A lun=21
source_updating=no create_vol=yes pool=test_pool
```

With the data migration defined through the script or batch job, an equivalent script or batch job to execute the data migrations then must be run, as shown in Example 9-4.

Example 9-4 Activate data migration batch file

```
xcli -m 10.10.0.10 dm_activate vol=MigVol_1
xcli -m 10.10.0.10 dm_activate vol=MigVol_2
xcli -m 10.10.0.10 dm_activate vol=MigVol_3
xcli -m 10.10.0.10 dm_activate vol=MigVol_4
xcli -m 10.10.0.10 dm_activate vol=MigVol_5
xcli -m 10.10.0.10 dm_activate vol=MigVol_6
xcli -m 10.10.0.10 dm_activate vol=MigVol_7
xcli -m 10.10.0.10 dm_activate vol=MigVol_8
xcli -m 10.10.0.10 dm_activate vol=MigVol_9
xcli -m 10.10.0.10 dm_activate vol=MigVol_10
```

9.6 Manually creating the migration volume

The local XIV volume can be pre-created before defining the data migration object. This option is not recommended because it is prone to manual calculation errors. This requires the size of the source volume on the non XIV storage system to be known in 512-byte blocks, as the two volumes (source and XIV volume) must be exactly the same size. Finding the actual size of a volume in blocks or bytes can be difficult, as certain storage devices do not show the exact volume size. This might require you to rely on the host operating system to provide the real volume size, but this is also not always reliable.

For an example of the process to determine exact volume size, consider ESS 800 volume 00F-FCA33 depicted in Figure 9-49 on page 298. The size reported by the ESS 800 web GUI is 10 GB, which suggests that the volume is 10,000,000,000 bytes in size (because the ESS 800 displays volume sizes using decimal counting). The AIX **`bootinfo -s hdisk2`** command reports the volume as 9,536 GiB, which is 9,999,220,736 bytes (because there are 1,073,741,824 bytes per GiB). Both of these values are too small. When the volume properties are viewed on the volume information panel of the ESS 800 Copy Services GUI, it correctly reports the volume as being 19,531,264 sectors, which is 10,000,007,168 bytes (because there are 512 bytes per sector). If we create a volume that is 19,531,264 blocks in size, this will be exactly reported as such. When the XIV automatically created a volume to migrate the contents of 00F-FCA33 it did create the volume as 19,531,264 blocks. Of the three information sources that were considered to manually calculate volume size, only one of them must have been correct. Using the automatic volume creation eliminates this uncertainty.

If you are confident that you determined the exact size, then when creating the XIV volume, choose the **Blocks** option from the Volume Size drop-down menu and enter the size of the XIV volume in blocks. If your sizing calculation is correct, an XIV volume is created that is the same size as the source (non XIV storage system) volume. Then, you can define a migration:

1. In the XIV GUI go to the floating menu **Remote** → **Migration**.
2. Right-click and choose **Define Data Migration** (Figure 9-15 on page 263). Make the following appropriate entries and selections, and then click **Define**:
   - Destination Pool: Choose the pool from the drop-down menu where the volume was created.
   - Destination Name: Chose the pre-created volume from the drop-down menu.
   - Source Target System: Choose the already defined non XIV storage system from the drop-down menu.

     Important: If the non XIV device is active/passive, the source target system must represent the controller (or service processor) on the non XIV device that currently owns the source LUN being migrated. This means that you must check from the non XIV storage, which controller is presenting the LUN to the XIV.

   - Source LUN: Enter the decimal value of the LUN as presented to the XIV from the non XIV storage system. Certain storage devices present the LUN ID as hex. The number in this field must be the decimal equivalent.
   - Keep Source Updated: Check this if the non XIV storage system source volume is to be updated with writes from the host. In this manner all writes from the host will be written to the XIV volume, and also the non XIV source volume until the data migration object is deleted.
3. Test the data migration object. Right-click to select the created data migration volume and choose **Test Data Migration**. If there are any issues with the data migration object the test fails reporting the issue that was found. See Figure 9-19 on page 265 for an example of the panel.

If the volume that you created is too small or too large you will receive an error message when you do a test data migration, as shown in Figure 9-25 on page 276. If you try and activate the migration you will get the same error message. You must delete the volume that you manually created on the XIV and create a new correctly sized one. This is because you cannot resize a volume that is in a data migration pair, and you cannot delete a data migration pair unless it has completed the background copy. Delete the volume and then investigate why your size calculation was wrong. Then create a new volume and a new migration and test it again.

Figure 9-25 XIV volume wrong size for migration

9.7 Changing and monitoring the progress of a migration

As mentioned previously, you can change the rate at which the data is copied from the source storage system to the XIV. This section described how and why these changes can be made.

9.7.1 Changing the synchronization rate

Only one tunable parameter determines the speed at which migration data is transferred between the XIV and defined targets. This setting can be tuned on a target-by-target setting. Two other tunable parameters apply to XIV Remote Mirroring (RM):

- max_initialization_rate

  The rate (in MBps) at which data is transferred between the XIV and defined targets. The default rate is 100 MBps and can be configured on a per-target basis, which means one target can be set to 100 MBps while another is set to 50 MBps. In this example, a total of 150 MBps (100 + 50) transfer rate is possible.

  In general, use caution when changing this value. The defaults are reasonable values that will balance migration activity and host access to the data being migrated. Consider the default of 100 MBps as a maximum rate, and use a lower value (50 MBps or less) for migrations from non-tier1 storage. Higher rates might actually be counter-intuitive and increase the migration time because of low-level protocol aborts and retries.

  If the transfer rate that you are seeing is lower than the initialization rate, this might indicate that you are exceeding the capabilities of the source storage system. If the migration is being performed online, consider dropping the initialization rate to a low number initially to ensure that the migration I/O does not interfere with other hosts using the non XIV storage system and that real-time read/writes can be satisfied. Then, slowly increase the number while checking to ensure that response times are not affected.

  If you set the max_initialization_rate to zero, then you will stop the background copy, but hosts will still be able to access all activated migration volumes.

- max_syncjob_rate

  This parameter (which is in MBps) is used in XIV remote mirroring for synchronizing mirrored snapshots. It is not normally relevant to data migrations. However, the max_initialization_rate cannot be greater than the max_syncjob_rate, which in turn cannot be greater than the max_resync_rate. In general, there is no reason to ever increase this rate.

- max_resync_rate

  This parameter (in MBps) is used for XIV remote mirroring only. It is not normally relevant to data migrations. This parameter defines the resync rate for mirrored pairs. After remotely mirrored volumes are synchronized, a resync is required if the replication is stopped for any reason. It is this resync where only the changes are sent across the link that this parameter affects. The default rate is 300 MBps. There is no minimum or maximum rate. However, setting the value to 400 or more in a 4 Gbps environment does not show any increase in throughput. In general, there is no reason to increase this rate.

Increasing the max_initialization_rate parameter *might* decrease the time required to migrate the data. However, doing so might affect existing production servers on the source storage storage system or affect real-time reads/writes. By increasing the rate parameters, more resources will be used to serve migrations and fewer for existing production I/O. An example is where the server stalls or takes longer than usual to boot and mount the disks (XIV disks in this case). By decreasing the sync rate, the background copy does not override the real-time reads and writes, and the server will boot as expected. Be aware of how these parameters affect migrations and also production.

The rate parameters can only be set using XCLI, not through the XIV GUI. The current rate settings are displayed by using the -x parameter, so run the **target_list -x** command. If the setting is changed, the change takes place as needed with immediate effect so there is no need to deactivate/activate the migrations (doing so blocks host I/O). Example 9-5 first displays the target list and then confirms the current rates using the -x parameter. The example shows that the initialization rate is still set to the default value (100 MBps). We then increase the initialization rate to 200 MBps. We can then observe the completion rate, as shown in Figure 9-21 on page 268, to see whether it has improved.

Example 9-5 Displaying and changing the maximum initialization rate

```
>> target_list
Name                    SCSI Type   Connected
Nextrazap ITSO ESS800   FC          yes
>> target_list -x target="Nextrazap ITSO ESS800"
<XCLIRETURN STATUS="SUCCESS" COMMAND_LINE="target_list -x target=&quot;Nextrazap ITSO ESS800&quot;">
  <OUTPUT>
            <target id="4502445">
                <id value="4502445"/>
                <creator value="xiv_maintenance"/>
                <creator_category value="xiv_maintenance"/>
                <name value="Nextrazap ITSO ESS800"/>
                <scsi_type value="FC"/>
                <xiv_target value="no"/>
                <iscsi_name value=""/>
                <connected value="yes"/>
                <port_list value="5005076300C90C21,5005076300CF0C21"/>
                <num_ports value="2"/>
                <system_id value="0"/>
                <max_initialization_rate value="100"/>
                <max_resync_rate value="300"/>
                <max_syncjob_rate value="300"/>
                <connectivity_lost_event_threshold value="30"/>
                <xscsi value="no"/>
            </target>
  </OUTPUT>
</XCLIRETURN>
>> target_config_sync_rates target="Nextrazap ITSO ESS800"  max_initialization_rate=200
Command executed successfully.
```

Important: Just because the initialization rate was increased does not mean that the actual speed of the copy increases. The source storage system or the SAN fabric might be the limiting factor. In addition, you might cause an impact to the host system by over-committing too much bandwidth to migration I/O.

9.7.2 Monitoring migration speed

You can use the Data Migration panel shown in Figure 9-21 on page 268 to monitor the speed of the migration. The status bar can be toggled between GB remaining, percent complete, and hours/minutes remaining. However, if you want to monitor the actual MBps, you must use an external tool. This is because the performance statistics displayed using the XIV GUI or using an XIV tool does not include data migration I/O (the back-end copy). They do, however, show incoming I/O rates from hosts using LUNs that are being migrated.

9.7.3 Monitoring the impact of migration on host latency

If you combine migration with an aggressive level of background copy, you might exceed the capabilities of the source storage, resulting in high host latency when accessing data that has not yet been migrated. You can monitor the performance of the server with the XIV Top tool that is included with the XIV GUI. Highlight either the host being migrated or the volumes being migrated and select **Latency** to see if the host is being negatively affected by the migration. If high latency (over 50 ms for instance) is being displayed and the end users are reporting slow host response times, then lower the `max_initialization_rate` parameter as detailed in 9.7.1, “Changing the synchronization rate” on page 276.

The XIV Top example shown in Figure 9-26 on page 279 represents a host with latency that is acceptable (less than 10 ms). If the background copy is causing high latency, lowering the max_initialization_rate should result in the latency also being lowered. You might need to tune the copy rate to find a point that satisfies the requirement to complete the migration in a timely fashion while delivering good performance to users. You should routinely monitor latency over the course of the migration window.

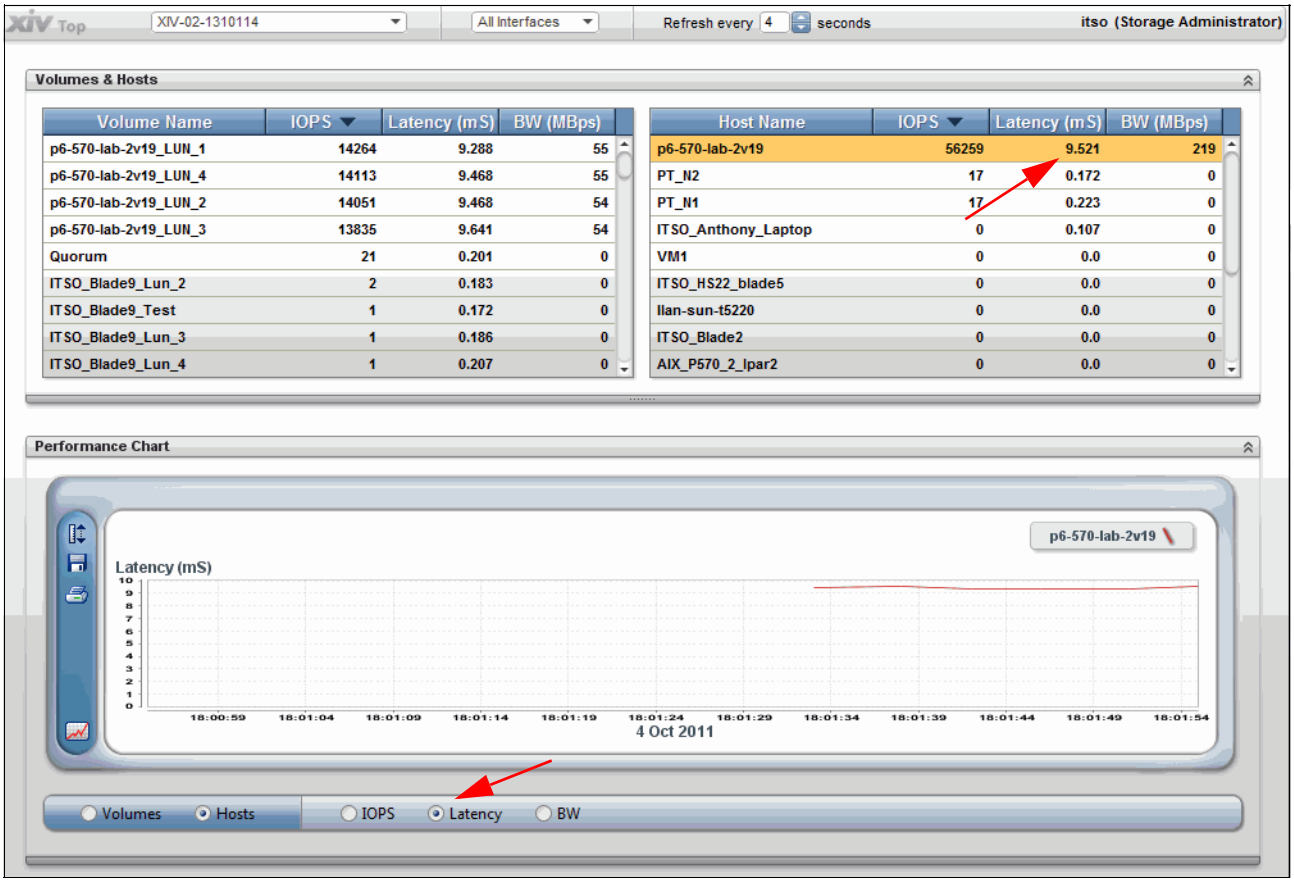


Figure 9-26 XIV Top showing host latency

9.7.4 Monitoring migration through the XIV event log

The XIV event log can be used to confirm when a migration started and finished. From the XIV GUI, go to **Monitor** → **Events**. In the Events panel, select **dm** from the **Type** drop-down menu, and then click **Filter**. Figure 9-27 displays the events for a single migration. In this example, the events must be read from bottom to top. You can sort the events by date and time by clicking the **Date** column in the Events panel.

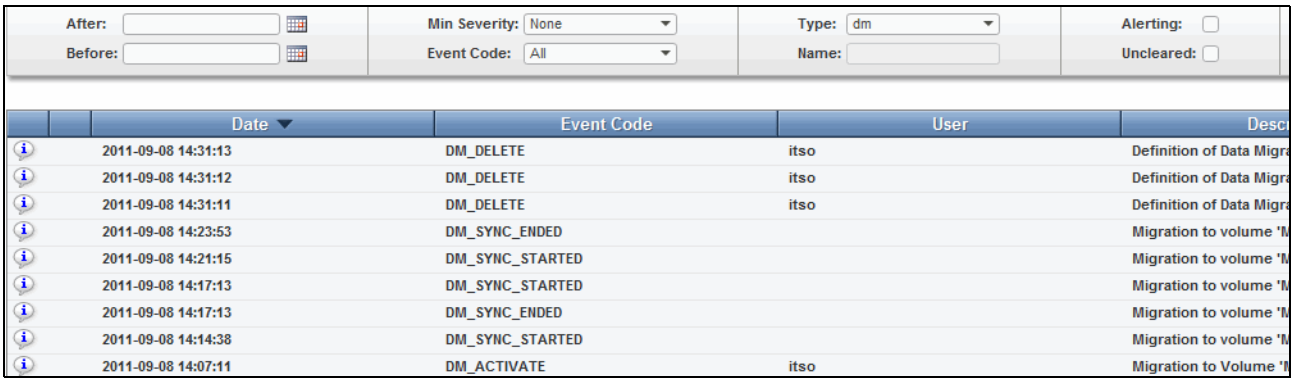

After:
Before:
Min Severity: None
Event Code: All
Type: dm
Name:
Alerting:
Uncleared:
Date | Event Code | User | Desc
2011-09-08 14:31:13 DM_DELETE itso Definition of Data Migra
2011-09-08 14:31:12 DM_DELETE itso Definition of Data Migra
2011-09-08 14:31:11 DM_DELETE itso Definition of Data Migra
2011-09-08 14:23:53 DM_SYNC_ENDED Migration to volume 'M
2011-09-08 14:21:15 DM_SYNC_STARTED Migration to volume 'M
2011-09-08 14:17:13 DM_SYNC_STARTED Migration to volume 'M
2011-09-08 14:17:13 DM_SYNC_ENDED Migration to volume 'M
2011-09-08 14:14:38 DM_SYNC_STARTED Migration to volume 'M
2011-09-08 14:07:11 DM_ACTIVATE itso Migration to Volume 'M

Figure 9-27 XIV Event GUI

9.7.5 Monitoring migration speed through the fabric

If you have a Brocade-based SAN, use the **portperfshow** command and verify the throughput rate of the initiator ports on the XIV. If you have two fabrics you might need to connect to two different switches. If multiple paths are defined between XIV and non XIV disk system, the XIV load balances across those ports. This means that you must aggregate the throughput numbers from each initiator port to see total throughput. Example 9-6 shows the output of the **portperfshow** command. The values shown are the combined send and receive throughput in MBps for each port. In this example port 0 is the XIV Initiator port and port 1 is a DS4800 host port. The max_initialization_rate was set to 50 MBps.

Example 9-6 Brocade portperfshow command

```
FB1_RC6_PDC:admin> portperfshow
0     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15  Total
================================================================================================
50m 50m  14m  14m 2.4m 848k 108k  34k   0  937k  0   27m 3.0m   0  949k 3.0m 125m
```

If you have a Cisco-based SAN, start Device Manager for the relevant switch and then select **Interface** → **Monitor** → **FC Enabled**.

9.7.6 Monitoring migration speed through the source storage system

The ability to display migration throughput varies by non XIV storage system. For example, if you are migrating from a DS4000 you could use the performance monitoring panels in the DS4000 System Manager to monitor the throughput. In the DS4000 System Manager GUI, go to **Storage Subsystem** → **Monitor Performance**. Display the volumes being migrated and the throughput for the relevant controllers. You can then determine what percentage of I/O is being generated by the migration process. In Figure 9-28 you can see that one volume is being migrated using a max_initialization_rate of 50 MBps. This represents the bulk of the I/O being serviced by the DS4000 in this example.

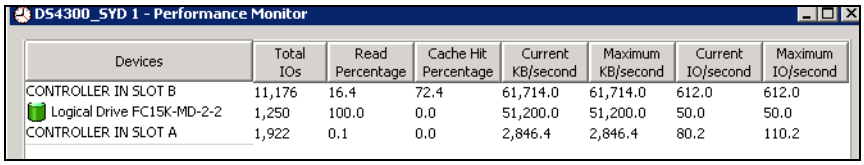


Figure 9-28 Monitoring a DS4000 migration

9.7.7 Predicting run time using actual throughput

Having determined the throughput in MBps, it is possible to use a simple graph to determine how many GB will be migrated every hour. For instance, in Figure 9-29 on page 281 you can see that if the background copy rate is 145 MBps then the copy rate will be approximately 500 GB per hour. If the rate is doubled to 290 MBps then approximately 1000 GB per hour can be migrated. Although a goal of migrating 1 TB per hour sounds attractive, it might be that the source storage can only sustain 50 MBps, in which case only 176 GB per hour can be migrated.

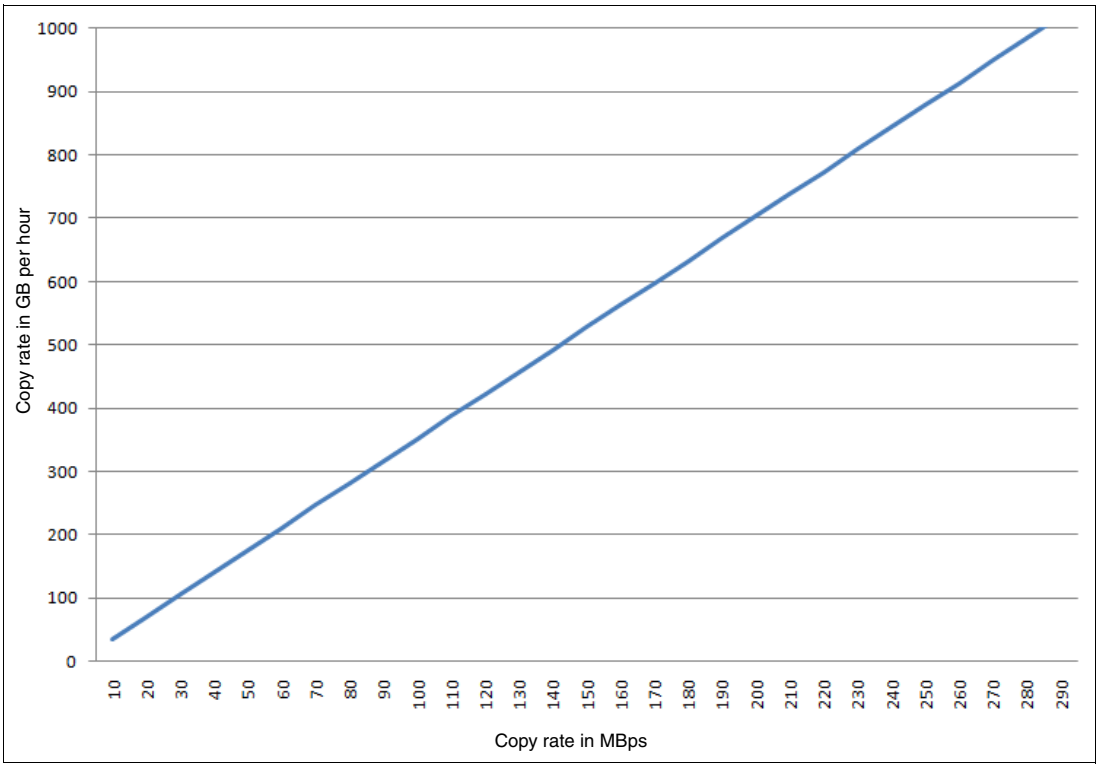


Figure 9-29 Copy rate per hour

9.8 Thick-to-thin migration

When the XIV migrates data from a LUN on a non XIV disk system to an XIV volume, it reads every block of the source LUN, regardless of contents. However, when it comes to writing this data into the XIV volume, the XIV only writes blocks that contain data. Blocks that contain only zeroes are not written and do not take any space on the XIV. This is called a *thick-to-thin* migration, and it occurs regardless of whether you are migrating the data into a thin provisioning pool or a regular pool.

While the migration background copy is being processed, the value displayed in the Used column of the Volumes and Snapshots panel drops every time that empty blocks are detected. When the migration is completed, you can check this column to determine how much real data was actually written into the XIV volume. In Figure 9-30 the used space on the Windows2003_D volume is 4 GB. However, the Windows file system using this disk shown in Figure 9-32 on page 283 shows only 1.4 GB of data. This might lead you to conclude, incorrectly, that the thick-to-thin capabilities of the XIV do not work.

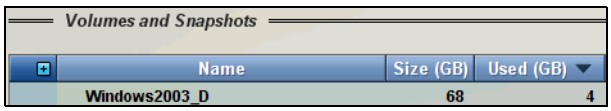


Figure 9-30 Thick-to-thin results

This situation happened because, when file deletions occur at a file-system level, the directory/file entry is removed but the data blocks are not. The file system reuses this effectively *free* space but does not write zeros over the old data (because doing so generates a large amount of unnecessary I/O). The end result is that the XIV effectively copies old and

deleted data during the migration. It must be clearly understood that this makes no difference to the speed of the migration, as these blocks have to be read into the XIV cache regardless of what they contain.

If you are not planning to use the thin provisioning capability of the XIV, this is not an issue. Only be concerned if your migration plan specifically requires you to be adopting thin provisioning.

Writing zeros to recover space

One way to recover space before you start a migration is to use a utility to write zeros across all free space. In a UNIX environment you could use a simple script like the one shown in Example 9-7 to write large empty files across your file system. You might need to run these commands many times to use all the empty space.

Example 9-7 Writing zeros across your file system

```
# The next command will write a 1 GB mytestfile.out
dd if=/dev/zero of=mytestfile.out bs=1000 count=1000000
# The next command will free the file allocation space
rm mytestfile.out
```

In a Windows environment, you can use a Microsoft utility known as **SDelete** to write zeros across deleted files. You can find this tool in the sysinternals section of Microsoft Technet:

http://technet.microsoft.com/en-us/sysinternals/bb897443.aspx

Important:

- If you choose to write zeros to recover space before the migration and the volume you are working with is mirrored by your non XIV storage device to a remote site, you might find these zeros also get mirrored to the remote site. This can add extra undesirable workload to your mirroring link, especially if the replication is synchronous. Check with the vendor who supplied your non XIV storage to see if it is able to avoid replicating updates that contain only zeros.
- If you instead choose to write zeros to recover space *after* the migration, you must initially generate large amounts of empty files, which might initially seem to be counter-productive. It might take up to three weeks for the used space value to decrease after the script or application is run. This is because recovery of empty space runs as a background task.

9.9 Resizing the XIV volume after migration

Because of the way that XIV distributes data, the XIV allocates space in 17 GB portions (which are exactly 17,179,869,184 bytes or 16 GiB). When creating volumes using the XIV GUI, this aspect of the XIV design becomes readily apparent when you enter a volume size and it gets rounded up to the next 17 GB cutoff.

If you chose to allow the XIV to determine the size of the migration volume, then you might find that a small amount of extra space is consumed for every volume that was created. Unless the volume sizes being used on the non XIV storage system were created in multiples of 16 GiB, then it is likely that the volumes automatically created by the XIV will reserve more XIV disk space than is actually made available to the volume.

The XIV volume properties of such an automatically created volume are shown in Figure 9-31. In this example the Windows2003_D drive is 53 GB in size, but the size on disk is 68 GB on the XIV.

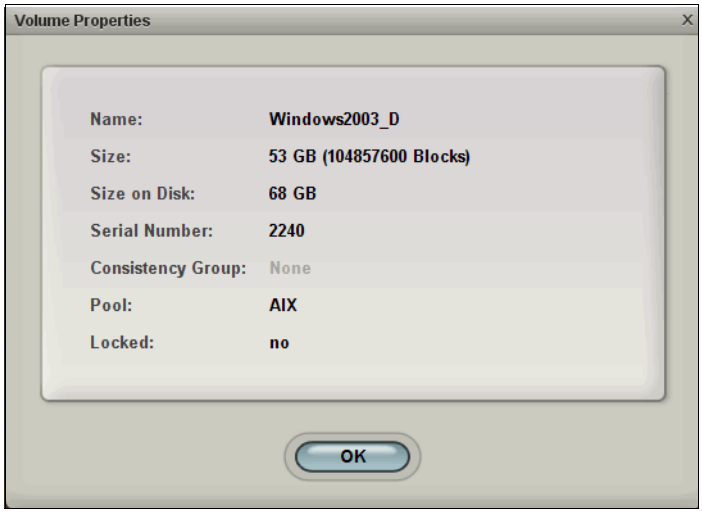


Figure 9-31 Properties of a migrated volume

This means that we can resize that volume to 68 GB (as shown in the XIV GUI) and make the volume 15 GB larger without effectively consuming any more space on the XIV. Figure 9-32 shows that the migrated Windows2003_D drive is 53 GB in size (53,678,141,440 bytes).

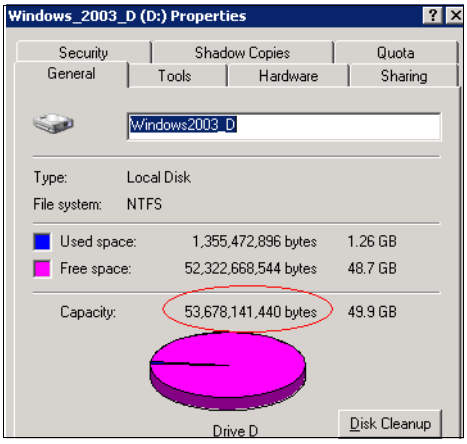


Figure 9-32 Windows D drive at 53 GB

To resize a volume go to the **Volumes** → **Volumes & Snapshots** panel, right-click to select the volume, then choose the **Resize** option. Change the sizing method drop-down from Blocks to **GB** and the volume size is automatically moved to the next multiple of 17 GB. You may also use XCLI commands, as shown in Example 9-8.

Example 9-8 Resize the D drive using XCLI

```
>> vol_resize vol=Windows2003_D size=68

Warning:   ARE_YOU_SURE_YOU_WANT_TO_ENLARGE_VOLUME Y/N: Y
Command executed successfully.
```

Because this example is for a Microsoft Windows 2003 basic NTFS disk, we can use the **diskpart** utility to extend the volume, as shown in Example 9-9.

Example 9-9 Expanding a Windows volume

```
C:\>diskpart
DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     C                NTFS   Partition     34 GB  Healthy    System
  Volume 4     D   Windows2003  NTFS   Partition     64 GB  Healthy

DISKPART> select volume 4
Volume 4 is the selected volume.
DISKPART> extend
DiskPart successfully extended the volume
```

Confirm that the volume has indeed grown by displaying the volume properties.

Figure 9-33 shows that the disk is now 68 GB (68,713,955,328 bytes).

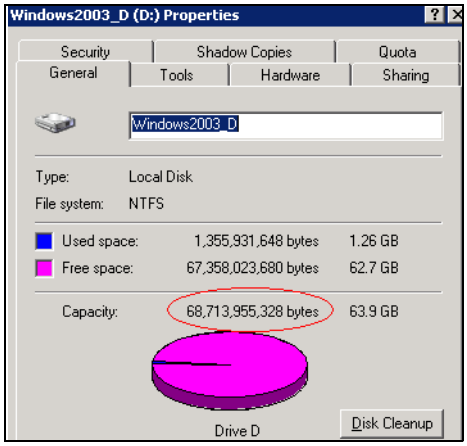


Figure 9-33 Windows 2003 D drive has grown to 64 GB

In terms of *when* to resize, a volume cannot be resized while it is part of a data migration. This means that the migration process must be completed and the migration for that volume must be deleted before the volume can be resized. For this reason you might choose to defer resizing until after the migration of all relevant volumes has been completed. This also separates the resize change from the migration change. Depending on the operating system using that volume, you might not get any benefit from doing this resizing.

9.10 Migrating XIV Generation 2 to XIV Gen3

There are various options for migrating data from XIV Generation 2 to XIV Gen3.

Gen3 to Gen3 migrations are better performed using XIV Hyper-Scale Mobility, which is also called *Online Volume Migration (OLVM)*, because no outage is required. For details, see *IBM Hyper-Scale in XIV Storage*, REDP-5053.

There are two storage-based methods for migrating data from Generation 2 to Gen3. The method used for migrating from Generation 2 to Gen3 depends on whether the migration is only local or includes XIV frames that are currently replicating. The methods are as follows:

- XIV Data Migration Utility (XDMU)
- XIV mirroring or replication

For local only, where the Generation 2 and Gen3 are in the same location, the best practice is the mirroring or replication method to perform the migration. This is because the data is moved to the new location (Gen3) before the server is moved. For those environments where data is already being replicated to a remote site, a hybrid solution might be the best approach, where a combination of XDMU and mirroring are used. This section describes each type and offers several recommendations.

Note: Replication target volumes (LUNs that are a target for a replication) cannot be migrated using XDMU. Likewise, migration target volumes (LUNs that are the target of a migration) cannot be replicated.

9.10.1 Generation 2 to Gen3 migration using XDMU

In this way, data is migrated from a Generation 2 system to a Gen3 system, using the migration process as described in this chapter. The Gen3 pulls or copies the data from Generation 2 to Gen3. The high-level process is to shut down the server, unzone the server from Generation 2, zone the server to Gen3, and then define and activate the data-migration (DM) volumes between the Generation 2 and Gen3. After the DMs are set up and activated, the Gen3 LUNs are allocated to the server at which time the server and applications can be brought online. This methodology between XIVs is no different than if the source storage system was a non XIV storage system. Gen3 treats the source XIV (Generation 2) the same way as though it were a DS3000 series or DS8000 series. The Gen3 is acting as a host, which is reading and writing (Keep Source Updated option) data to or from the Generation 2.

Note: Do not forget to define the Gen3 as a host on the Generation 2 and add the proper WWN ports (initiator ports from the Gen3). Until this is done, the migration connectivity links on the Gen3 will not be running (shows as green in the GUI). SCSI protocol dictates that LUN-0 must exist on the target/storage system. Until the Gen3 WWN ports are defined on the Generation 2, the Gen3 cannot see the Generation 2 LUN-0 and complete the connection. All storage systems present a pseudo LUN-0 unless a real (actual data LUN) LUN-0 is provided.

9.10.2 Generation 2 to Gen3 migration using replication

By using replication to migrate the data, the data is moved to the Gen3 before the server is moved (all the data must reside on the Gen3 before the server is moved). Replication is set up between XIV Generation 2 and XIV Gen3, and the data is replicated to the new XIV Gen3 while the server and applications remain online and still attached to the Generation 2. After

the volumes being replicated/migrated are synchronized, the server is shut down, unzoned from the XIV Generation 2, and zoned to the XIV Gen3. After the Gen3 LUNs are allocated to the server, the server and applications are brought online. As with any replication or migration, this is done on a LUN-by-LUN or server-by-server basis. There is no need to replicate or migrate all the LUNs at the same time. Use synchronous replication rather than asynchronous. Because the XIV frames are typically in the same data center, latency is not an issue. Asynchronous replication can be used, but is slightly more involved and requires snapshot space on both Generation 2 XIV and Gen3 XIV. See Chapter 6, “Asynchronous remote mirroring” on page 149 for replication details.

9.10.3 Generation 2 to Gen3 migration in multi-site environments

In this section, we discuss solutions for environments where XIV replication or mirroring is already in use. As noted previously, replication target volumes (LUNs that are a target for replication) cannot be migrated using XDMU. Likewise, migration target volumes (LUNs that are the target of a migration) cannot be replicated. Therefore, for LUNs that are already being replicated, a hybrid approach might be required in the following cases:

- Source Generation 2 being replaced, but the remote Generation 2 is not.
- Both source and remote Generation 2 systems are being replaced.

Considerations

When migrating data from Generation 2 to Gen3 in multi-site environments, several items must be considered and addressed:

- DR requirements: What is the DR outage window? For example, how long can the remote site’s data be out of sync. Both the hybrid solutions listed above require some DR outage window where the data between the source and remote sites will not be in sync. This is because during the migration process, at some point, the old mirroring pair is deleted in order to re-establish it with the new Gen3. Host-based migrations, such as Logical Volume Managers (LVM), Oracle Automatic Storage Management (ASM), Storage vMotion (SVM), and so on, is the only way to perform migrations where no DR outage is required. Using host-based migrations, the DR data is always current and available.
- Replication strategy: Is the current environment using synchronous or asynchronous replication? In some cases, the source and remote site (DR) can be resynchronized using offline init (truck mode). This is where only the changes that have occurred since the replication pair was deleted are replicated. If the current replication strategy is asynchronous, then offline init can easily be used to re-establish the replication pair. If synchronous is the current replication, then offline init can be used if the source and remote Gen3 are running XIV code 11.4 or later. Otherwise a full resynchronization is required.
- Network bandwidth and latency: What is the replication network’s bandwidth and latency. This is an important consideration because a full resynchronization might be required between the source and DR site, while existing mirrored pairs remain operational.

Source Generation 2 replacement only

In this scenario, the source/primary site Generation 2 is being replaced with a new Gen3 while the remote/DR site Generation 2 remains and is not being replaced. The data at the source site will be migrated to the new Gen3 using XDMU with keeping the source updated. Replication will continue between the source and DR Generation 2. Using the methodology, data written by the server to the Gen3 will be synchronously written to the Generation 2 (using XDMU, keep source update), which in turn replicates to the remote site Generation 2. After the data is moved to the new Gen3, the mirroring pair is deleted between the source and DR Generation 2. In an asynchronous environment, the newly migrated Gen3 LUNs can now

be mirrored with the original DR Generation 2 target LUNs using offline init. Only the changes that occurred since the migration was deleted are replicated between the source site Gen3 and the DR Generation 2. In a synchronous environment, a full synch is required between the source Gen3 and DR Generation 2 because the Generation 2 does not support synchronous offline init. See Figure 9-34 and Figure 9-35.

Note: In these two figures, XIV Generation 2 is depicted as Gen2.

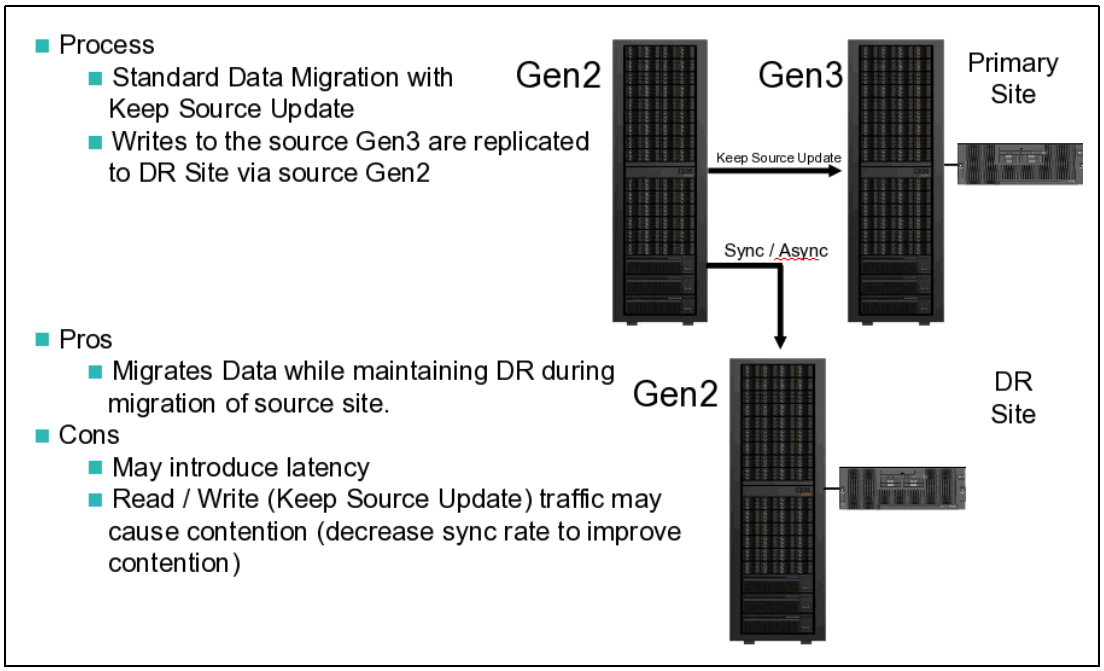

Process
Standard Data Migration with Keep Source Update
Writes to the source Gen3 are replicated to DR Site via source Gen2
Gen2
Gen3
Primary Site
Keep Source Update
Sync / Async
Pros
Migrates Data while maintaining DR during migration of source site.
Cons
May introduce latency
Read / Write (Keep Source Update) traffic may cause contention (decrease sync rate to improve contention)
Gen2
DR Site

Figure 9-34 Replace source Generation 2 only; Phase 1

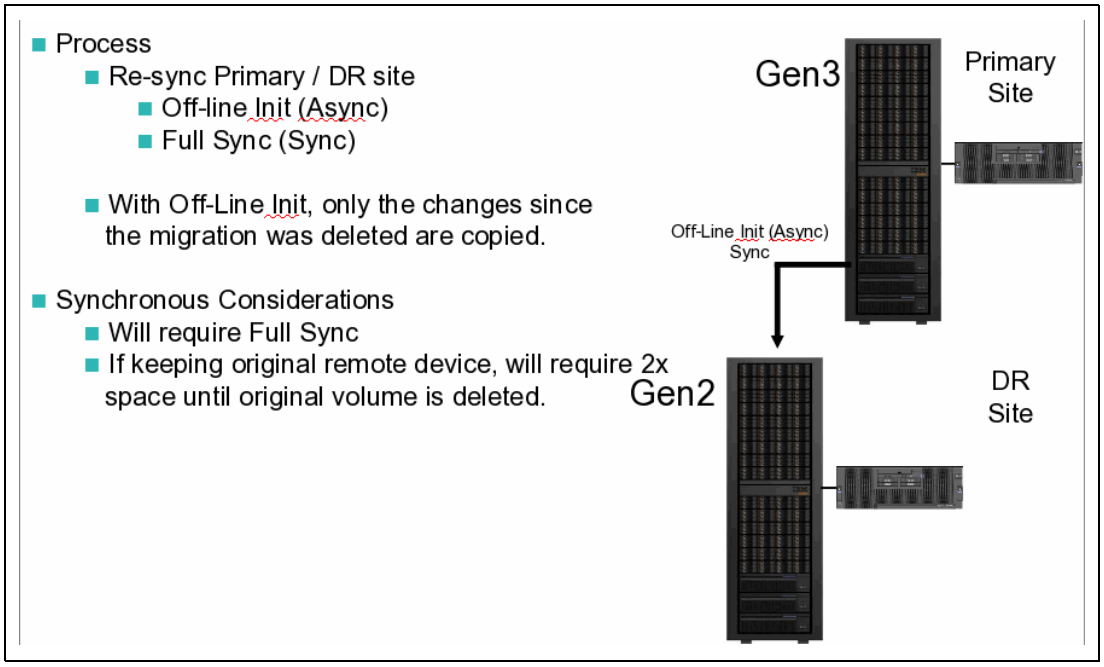


Figure 9-35 Replace source Generation 2 only; Phase 2

Source and DR Generation 2 being replaced with Gen3

In this scenario, both the source and DR Generation 2s are being replaced with Gen3s. As with the previous scenario, replication is maintained as long as possible while the data is migrated from the source Generation 2 to the source Gen3. The data at the source site will be migrated to the new Gen3 using XDMU, with keep source updated option. Replication continues between the source and DR Generation 2. Using the methodology, data written by the server to Gen3 is synchronously written to Generation 2 (using XDMU keep source updated option), which in turn replicates to the remote site Generation 2. After the data is moved to the new Gen3, the mirroring pair is deleted between the source and DR Generation 2.

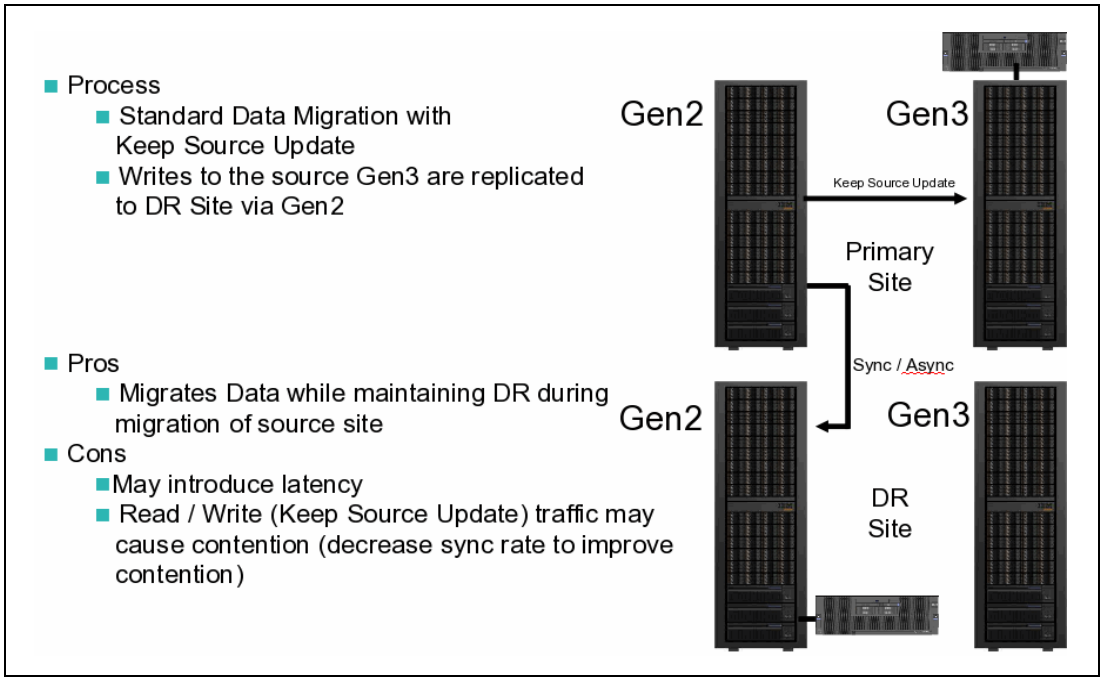

Process
Standard Data Migration with Keep Source Update
Writes to the source Gen3 are replicated to DR Site via Gen2
Pros
Migrates Data while maintaining DR during migration of source site
Cons
May introduce latency
Read / Write (Keep Source Update) traffic may cause contention (decrease sync rate to improve contention)
Gen2
Gen3
Keep Source Update
Primary Site
Sync / Async
Gen2
Gen3
DR Site

Figure 9-36 Source and DR Generation 2 being replaced; Phase 1

After this above migration is performed, you may choose one of two options:

- Option 1 (sync DR Generation 2 to DR Gen3): The idea with this option is to set up the scenario where the source Gen3 and DR Gen3 can be resynchronized using offline init, thereby minimizing the amount of data that must be synchronized between the source and DR Gen3 across the WAN. With this option, the DR Generation 2 is replicated to the new DR Gen3. After the DR Generation 2 and Gen3 are in sync, the mirrored pairs are deleted and an offline init is performed between the source Gen3 and the DR Gen3 replication pairs. This option adds the extra step of synchronizing the DR Generation 2 to the DR Gen3, but reduces WAN traffic because only the changes from the time the source migration is deleted are replicated. This option can be used for all asynchronous environments and can be used in synchronous environments where the source and DR Gen3 frames are running 11.4 code or later. See Figure 9-37 on page 289. You may skip this option if you want a full resynchronization.
- Option 2 (full resync): Use this option if you want or require a full synchronization between the source Gen3 and the DR Gen3. This option can be used at any time, but is specifically meant for synchronous environments using XIV code 11.3 and earlier. In those environments, a full sync is required. With XIV code 11.4 and later, synchronous offline init is available and a full sync is no longer required. See Figure 9-38 on page 289.

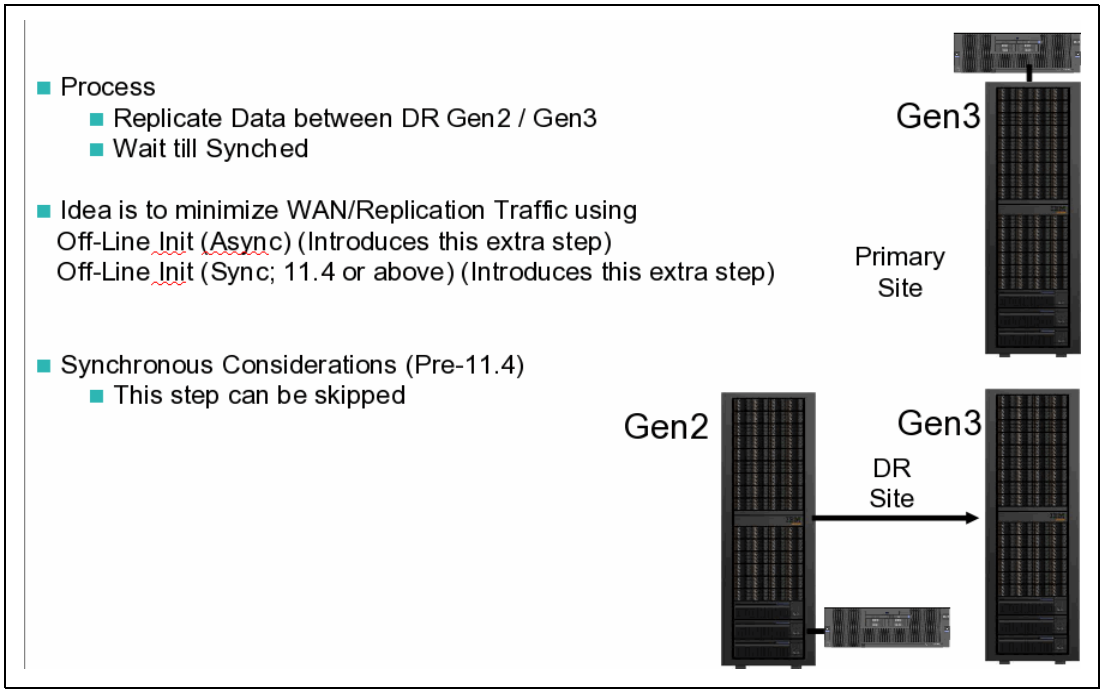


Figure 9-37 Source and DR Generation 2 being replaced; Phase 2

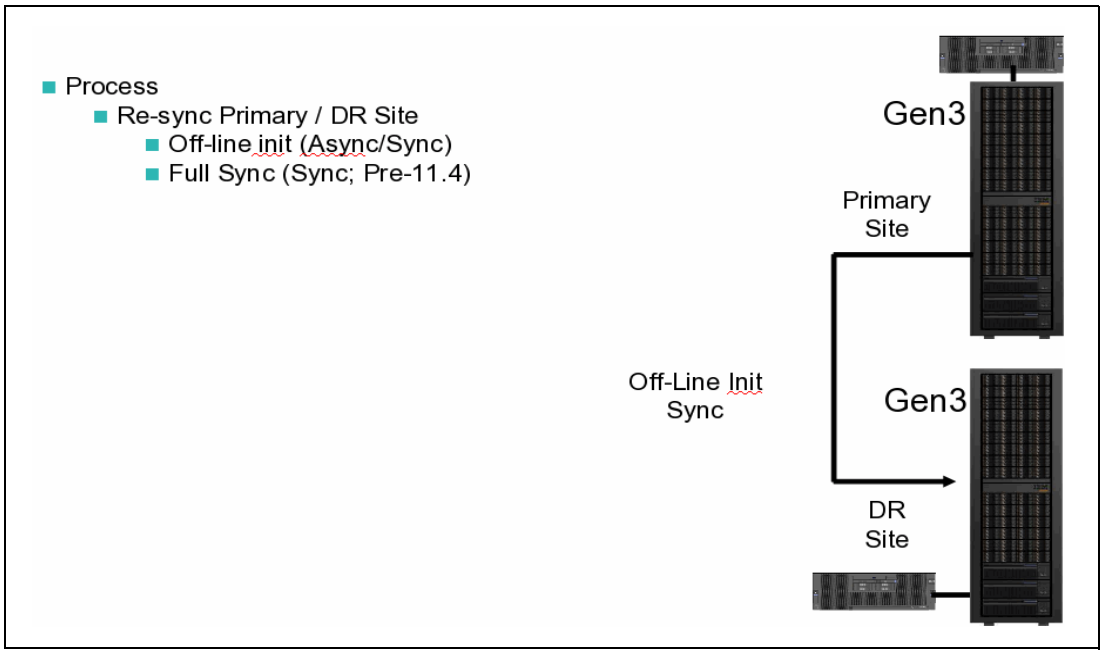


Figure 9-38 Source and DR Generation 2 being replaced; Phase 3

9.10.4 Server- based migrations

Software or host-based migration is the only option to migrate data from Generation 2 to Gen3 with no server outage or DR outage. This option uses server-based applications to migrate the data such as Logical Volume Managers (LVM), or other applications such as VMware Storage vMotion (SVM) or Oracle ASM. With this methodology, new volumes from the new source Gen3 are allocated to the server. The existing Generation 2 LUNs are kept in place and LVM is used to mirror or migrate the data from the Generation 2 LUNs to the new Gen3 LUNs. In this section, we discuss two scenarios where applications can be used to migrate from Generation 2 to Gen3 without interruption.

- Local only migration
- Environments with replication

Note: Because server-based migrations can also be used to perform LUN consolidation, consider LUN consolidation considered during this process. XIV is much more efficient with fewer larger LUNs. This is because XIV's large cache and advanced look-ahead algorithms are better utilized with larger LUNs.

Local only migration

This is where a Gen3 XIV is replacing a Generation 2 with no replication. Here new LUNs from the Gen3 are allocated to an existing server and are then configured into the application whether it is LVM, Oracle ASM, or another application where data between two LUNs can be mirrored. After the old Generation 2 LUNs are synchronized with the new Gen3 LUNs, the mirrors are broken and the old Generation 2 LUNs removed from the configuration, unmapped or unallocated, and unzoned from the server. See Figure 9-39.

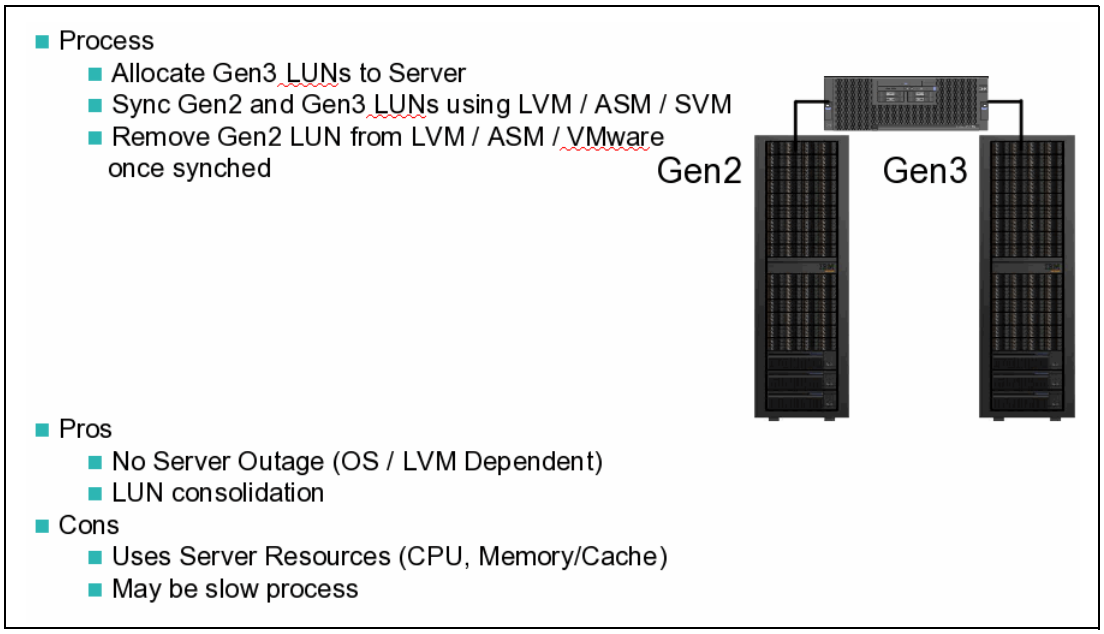

Process
Allocate Gen3 LUNs to Server
Sync Gen2 and Gen3 LUNs using LVM / ASM / SVM
Remove Gen2 LUN from LVM / ASM / VMware once synched
Gen2
Gen3
Pros
No Server Outage (OS / LVM Dependent)
LUN consolidation
Cons
Uses Server Resources (CPU, Memory/Cache)
May be slow process

Figure 9-39 Server-based migrations; local only, no replication

Environments with replication

This section examines several scenarios of migrating Generation 2 to Gen3, using server-based migrations in those environments where replication is already in place.

Source Generation 2 replacement only (Option 1:DR outage)

In this scenario, the source/primary site Generation 2 is being replaced with a new Gen3 while the remote/DR site Generation 2 remains and is not being replaced. The data at the source site is migrated to the new Gen3 using server-based migrations. Replication continues between the source and DR Generation 2. Using the methodology, data written by the server will be written to both the Generation 2 and Gen3 using LVM mirroring/ASM/SVM. Since the source Generation 2 is still replicating to the DR Generation 2, the data is replicated to the DR site. See Figure 9-40.

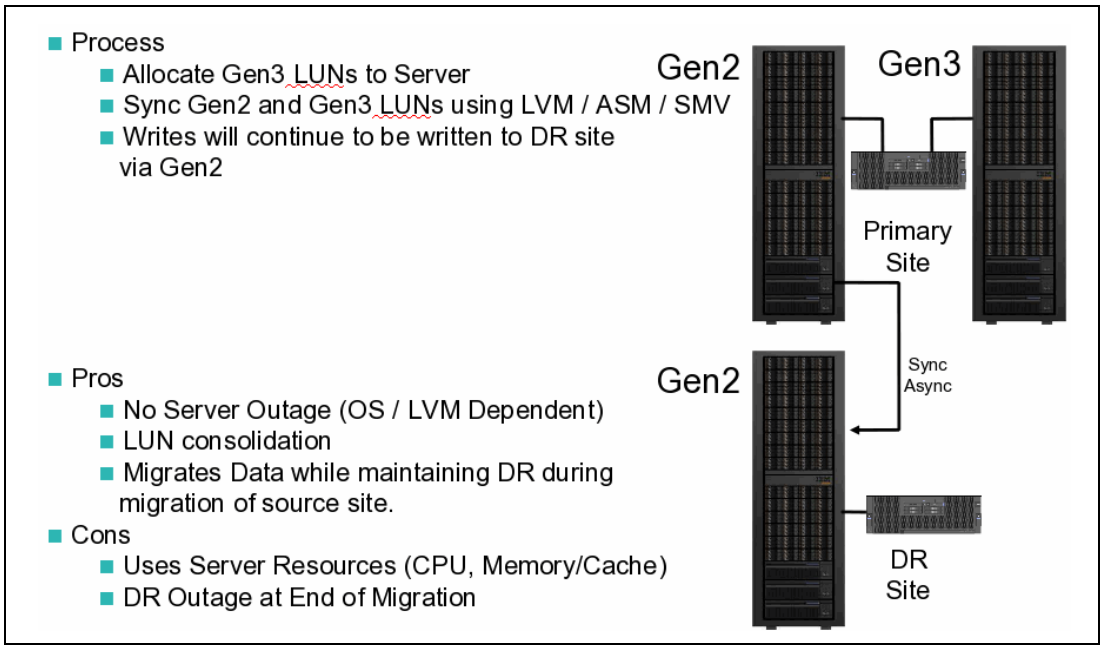


Figure 9-40 Replace source Generation 2 only; Phase 1

After the data is migrated to Gen3 and the LUNs are in sync, the old Generation 2 LUNs are removed from the configuration, and unmapped and unzoned from the server. At this point, the new Gen3 LUNs can be resynchronized with DR site Generation 2 using offline init in asynchronous environments. In synchronous environments, a full synchronization is required. See Figure 9-41 on page 292. In synchronous environments, the client might decide to keep the original Generation 2 target volume as a gold copy until the new mirrored pair between the source Gen3 and DR Generation 2 are in sync. In doing so, understand that until the original DR Generation 2 target volume is deleted, twice the space is required on the DR Generation 2.

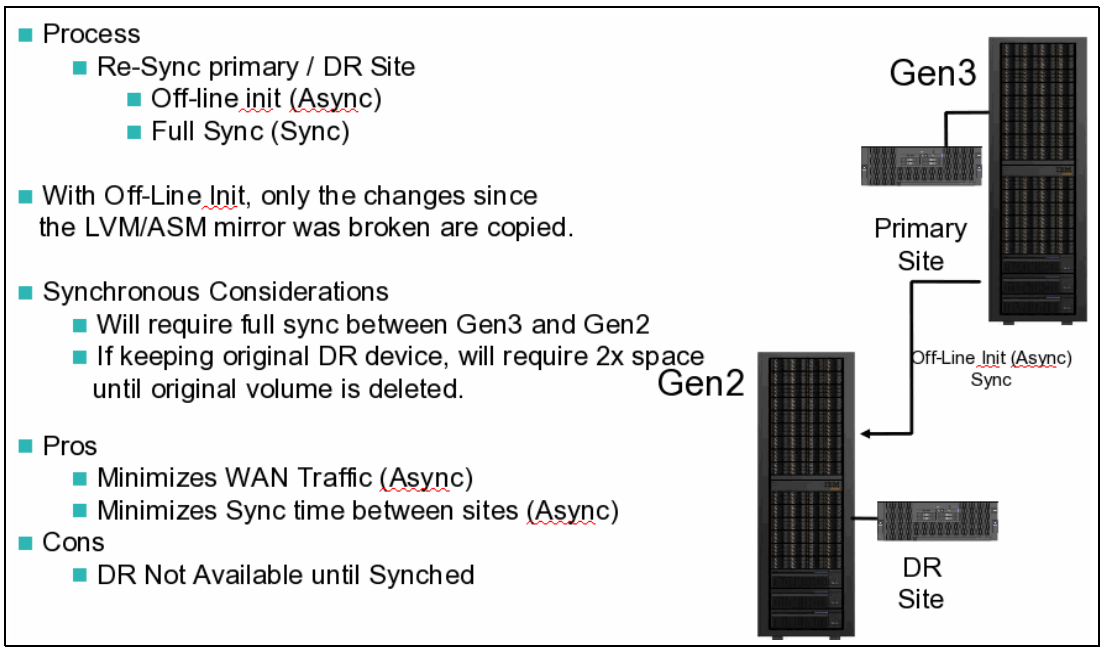

Process
Re-Sync primary / DR Site
Off-line init (Async)
Full Sync (Sync)
With Off-Line Init, only the changes since the LVM/ASM mirror was broken are copied.
Synchronous Considerations
Will require full sync between Gen3 and Gen2
If keeping original DR device, will require 2x space until original volume is deleted.
Pros
Minimizes WAN Traffic (Async)
Minimizes Sync time between sites (Async)
Cons
DR Not Available until Synched
Gen3
Primary Site
Off-Line Init (Async) Sync
Gen2
DR Site

Figure 9-41 Replace source Generation 2 only; Phase 2

Source Generation 2 replacement only (Option 2: DR-maintained)

This scenario is much like the previous one except the difference is that a mirrored pair will be created between source Gen3 and DR Generation 2 before the data is migrated by using host based migrations at the source. The advantage of this option is that there is no DR outage; a current DR copy of the data is always available. While the source data is being migrated, the source and DR Generation 2s remain in sync while the source Gen3 and DR Generation 2 mirrored pairs are synchronizing. After the migration is complete, the source Generation 2 volumes are removed from the server configuration, and unallocated and unzoned from the server. At this point, the mirrored pairs between the source and DR Generation 2 can be deleted.

This scenario requires twice the space on the DR Generation 2 because the original DR target LUN will not be deleted until the source migration is complete and the source Gen3 and DR Generation 2 are in sync. In asynchronous environments, carefully consider snapshot space in storage pools on the DR Generation 2 until the original DR target LUNs are deleted. Also note in this methodology that server write operations are written twice across the replication network: once between the source and DR Generation 2s and again between the source Gen3 and DR Generation 2. See Figure 9-42 on page 293 and Figure 9-43 on page 293.

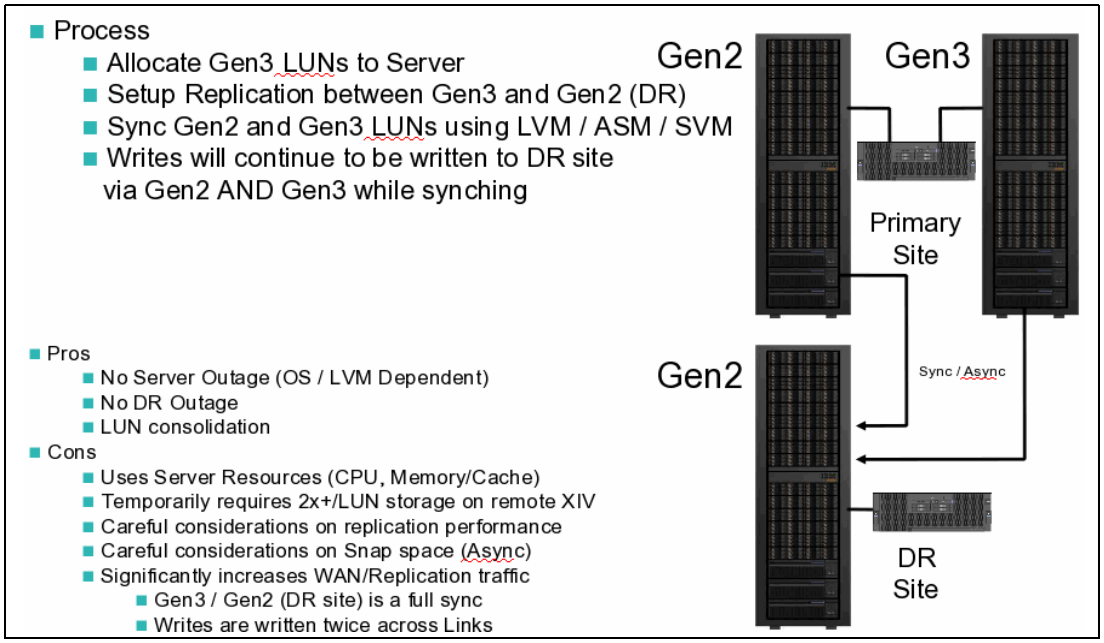

Process
Allocate Gen3 LUNs to Server
Setup Replication between Gen3 and Gen2 (DR)
Sync Gen2 and Gen3 LUNs using LVM / ASM / SVM
Writes will continue to be written to DR site via Gen2 AND Gen3 while synching
Pros
No Server Outage (OS / LVM Dependent)
No DR Outage
LUN consolidation
Cons
Uses Server Resources (CPU, Memory/Cache)
Temporarily requires 2x+/LUN storage on remote XIV
Careful considerations on replication performance
Careful considerations on Snap space (Async)
Significantly increases WAN/Replication traffic
Gen3 / Gen2 (DR site) is a full sync
Writes are written twice across Links
Gen2
Gen3
Primary Site
Sync / Async
Gen2
DR Site

Figure 9-42 Replace source Generation 2 only; Phase 1

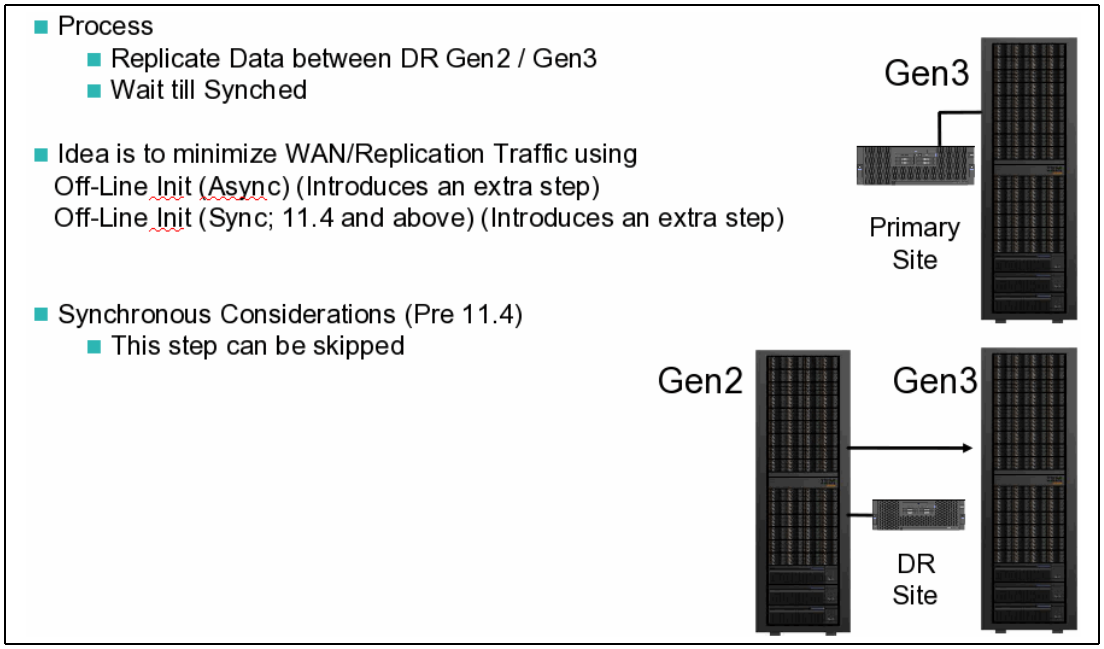


Figure 9-43 Replace source Generation 2 only; Phase 2

Source and DR Generation 2 being replaced (Option 1: no DR outage)

In this scenario, both the source and DR Generation 2s are being replaced with Gen3s. The replication between the source and DR Generation 2 stays in place until the source migration is complete and the source and DR Gen3 volumes are in sync. This is done by allocating new Gen3 LUNs/volumes to the source server/application and letting the application (LVM, ASM, SVM) perform the migrations. At the same time, create a mirrored pair between the source and DR Gen3. While the data is migrating between the source Generation 2 and Gen3, the data is being replicated between the source and DR Generation 2 and the source and DR Gen3 volumes are synchronizing. After the migration is complete between the source

Generation 2 and Gen3, the Generation 2 LUNs are removed from the configuration, unallocated, and unzoned from the server. This method can provide 100% DR data availability; DR data is always available. With this methodology, server writes are written twice across the replication network: once between the source and DR Generation 2s, and again between the source and DR Gen3. See Figure 9-44 and Figure 9-45

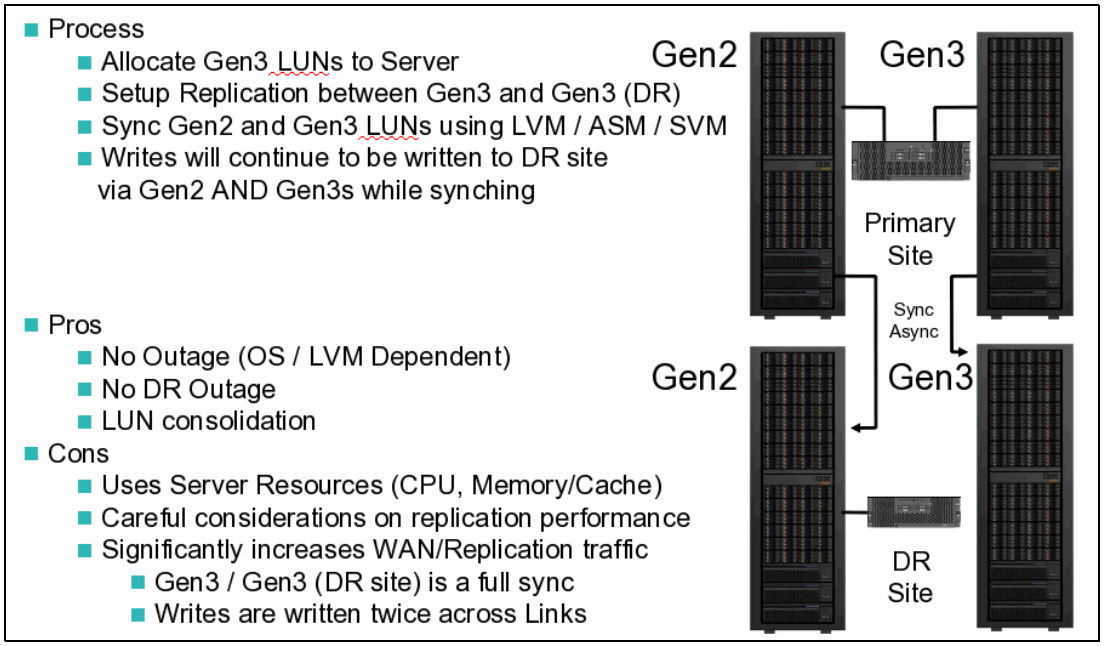

Process
Allocate Gen3 LUNs to Server
Setup Replication between Gen3 and Gen3 (DR)
Sync Gen2 and Gen3 LUNs using LVM / ASM / SVM
Writes will continue to be written to DR site via Gen2 AND Gen3s while synching
Pros
No Outage (OS / LVM Dependent)
No DR Outage
LUN consolidation
Cons
Uses Server Resources (CPU, Memory/Cache)
Careful considerations on replication performance
Significantly increases WAN/Replication traffic
Gen3 / Gen3 (DR site) is a full sync
Writes are written twice across Links
Gen2
Gen3
Primary Site
Sync Async
Gen2
Gen3
DR Site

Figure 9-44 Source and DR Generation 2 being replaced; Phase 1

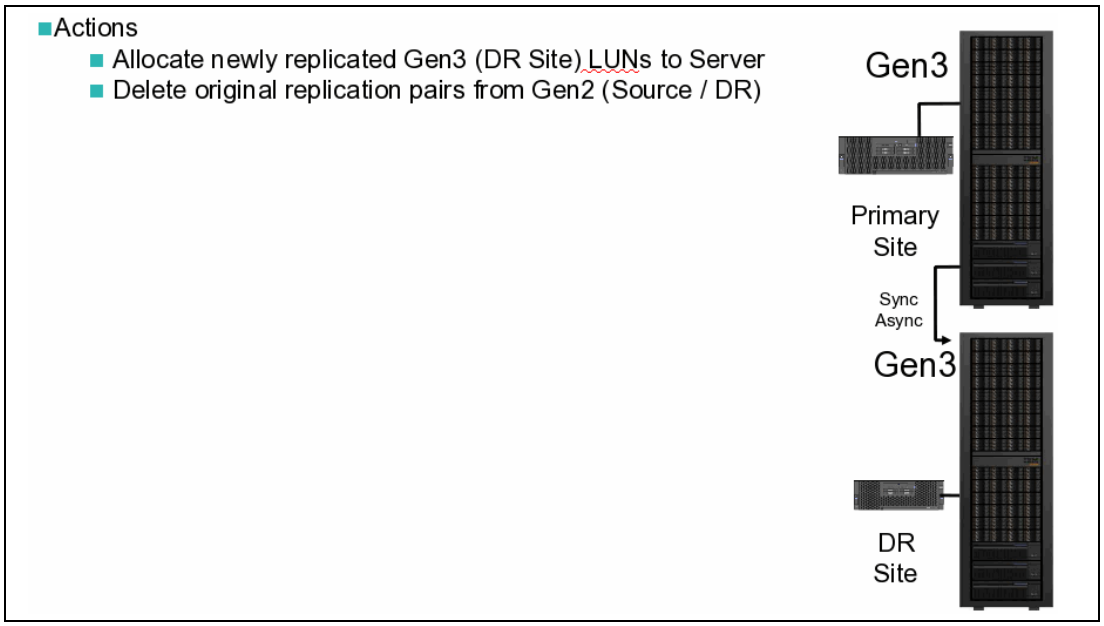


Figure 9-45 Source and DR Generation 2 being replaced; Phase 2

Source and DR Generation 2 being replaced (Option 2: DR outage)

An alternative to the previous methodology where the replication network is highly utilized or has little bandwidth available, is to add an extra step of replicating the DR Generation 2 LUNs to the DR Gen3 and then performing an offline init between the source and DR Gen3s. This can be done instead of performing a full synchronization between the source and DR Gen3s. However, with this alternative method, a DR data availability outage will occur for a period of time: until the source and DR Gen3 LUNs are in sync and a consistent state. See Figure 9-46 through Figure 9-48 on page 296.

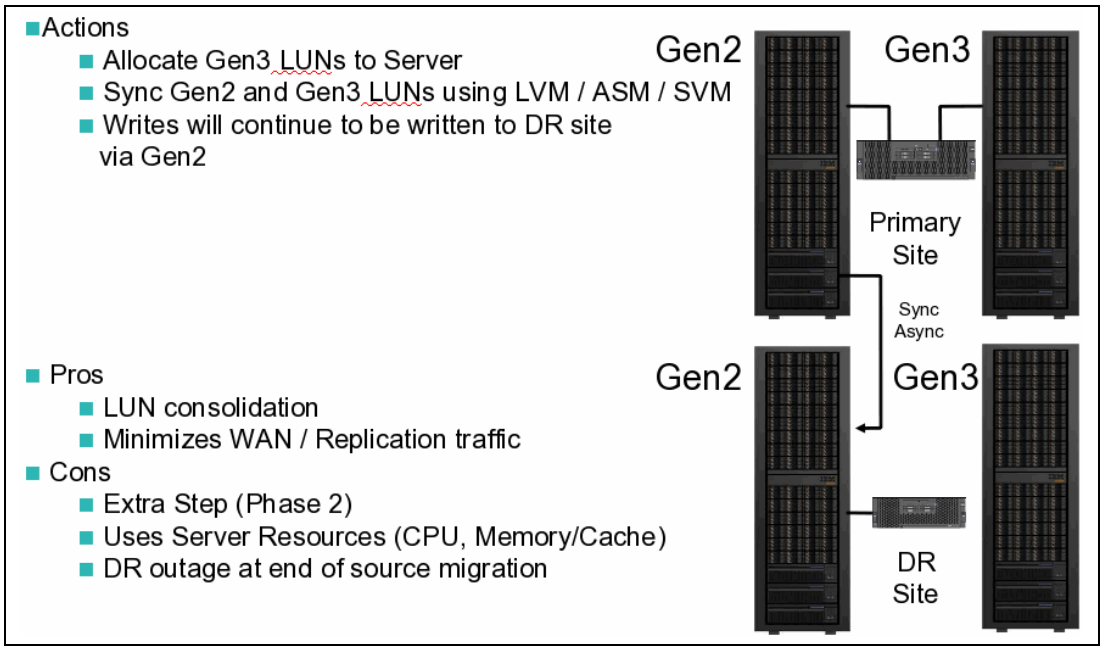


Figure 9-46 Source and DR Generation 2 being replaced (Option 2: DR outage); Phase 1

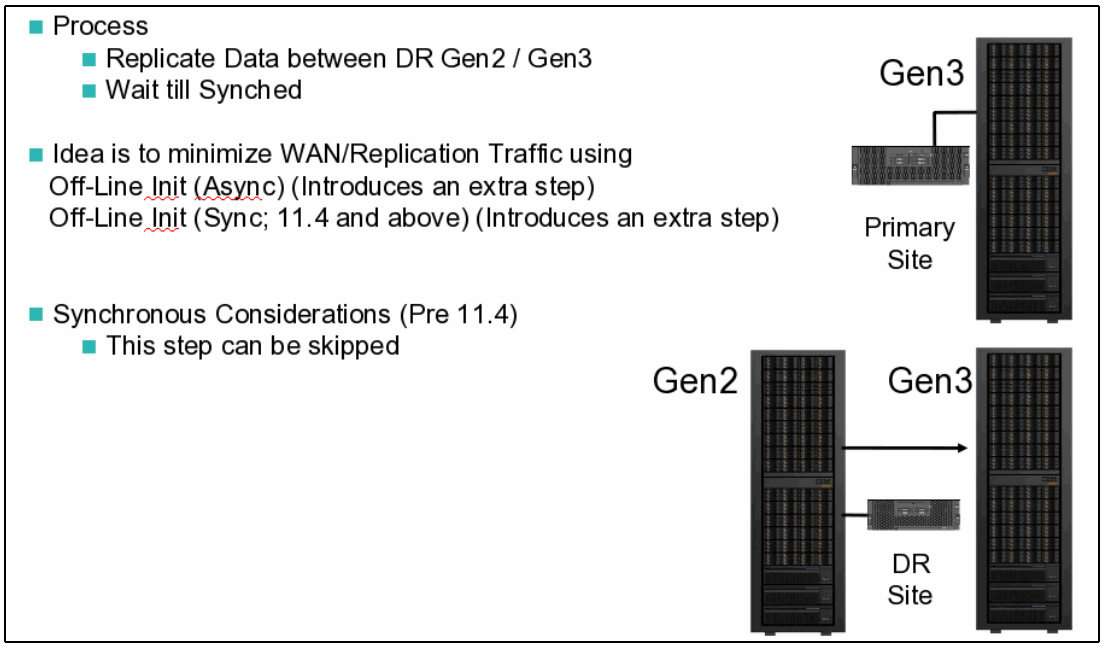


Figure 9-47 Source and DR Generation 2 being replaced (Option 2: DR outage); Phase 2

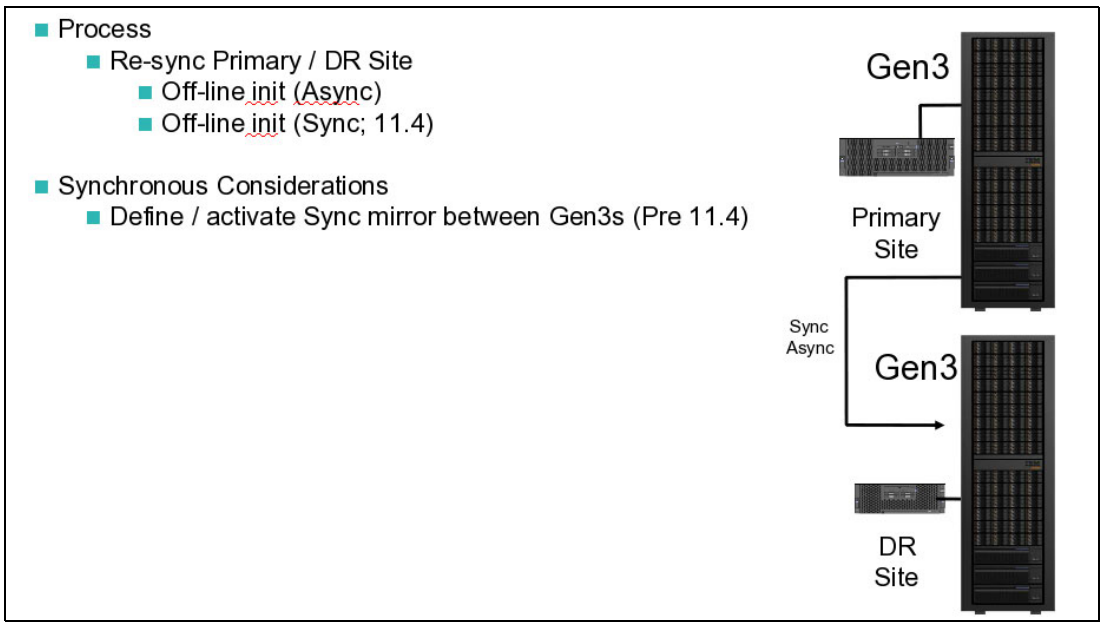


Figure 9-48 Source and DR Generation 2 being replaced (Option 2: DR outage); Phase 3

9.11 Troubleshooting

This section lists common errors that are encountered during data migrations using the XIV data migration facility.

9.11.1 Target connectivity fails

The connections (link line) between the XIV and non XIV disks system on the migration connectivity panel remain colored red or the link shows as *down*. This can happen for several reasons; do the following tasks:

- On the Migration Connectivity panel, verify that the status of the XIV initiator port is *OK (Online)*. If not, check the connections between the XIV and the SAN switch.
- Verify that the Fibre Channel ports on the non XIV storage system are set to target, enabled, and online.
- Check whether SAN zoning is incorrect or incomplete. Verify that SAN fabric zoning configuration for XIV and non XIV storage system are active.
- Check SAN switch nameserver that both XIV ports and non XIV storage ports have logged in correctly. Verify that XIV and non XIV are logged into the switch at the correct speed.
- Determine if the XIV WWPN is properly defined to the non XIV storage system target port. The XIV WWPN must be defined as a Linux or Windows host.
  - If the XIV initiator port is defined as a Linux host to the non XIV storage system, change the definition to a Windows host. Delete the link (line connections) between the XIV and non XIV storage system ports, and redefine the link. This depends on the storage device and is caused by how the non XIV storage system presents a pseudo LUN-0 if a real volume is not presented as LUN 0.
  - If the XIV initiator port is defined as a Windows host to the non XIV storage system, change the definition to a Linux host. Delete the link (line connections) between the XIV and non XIV storage system ports and redefine the link. This depends on the

storage device and is caused by how the non XIV storage system presents a pseudo LUN-0 if a real volume is not presented as LUN 0.

If these procedures for Linux and Windows are not successful, assign a real disk/volume to LUN 0 and present it to the XIV. The volume assigned to LUN-0 can be a small unused volume or a real volume that will be migrated.

- Take the XIV Fibre Channel port offline and then online again. Go to the Migration Connectivity panel, expand the connectivity of the target by clicking the link between XIV and the target system, highlight the port in question, right-click, and select **Configure**. Click **No** in the second row drop-down menu (Enabled) and click **Configure**. Repeat the process, choosing **Yes** for Enabled.
- Change the port type from Initiator to Target and then back to Initiator. This forces the port to completely reset and reload. Go to the Migration Connectivity panel, expand the connectivity of the target by clicking the link between the XIV and target system, highlight the port in question, right-click, and select **Configure**. Select **Target** in the third row drop-down menu (Role) and click **Configure**. Repeat the process, choosing **Initiator** for the role.

9.11.2 Remote volume LUN is unavailable

This error typically occurs when defining a DM and the LUN ID specified in the Source LUN field is not responding to the XIV. This can occur for several reasons:

- The LUN ID (host LUN ID or SCSI ID) specified is not allocated to the XIV on the ports identified in the target definition (using the Migration Connectivity panel). You must log on to the non XIV storage system to confirm.
- The LUN ID is not allocated to the XIV on all ports specified in the target definition. For example, if the target definition has two links from the non XIV storage system to the XIV, the volume must be allocated down both paths using the same LUN ID. The XIV looks for the LUN ID specified on the first defined path. If it does not have access to the LUN it will fail even if the LUN is allocated down the second path. The LUN must be allocated down all paths as defined in the target definition. If two links are defined from the target (non XIV) storage device to the XIV, then the LUN must be allocated down both paths.
- The LUN ID is incorrect. Do not confuse a non XIV storage system's internal LUN ID with the SCSI LUN ID (host LUN ID) that is presented to the XIV. This is a common oversight. The source LUN must be the LUN ID (decimal) as presented to the XIV.
- The Source LUN ID field is expecting a decimal number. Certain vendors present the LUN ID in hex. This must be translated to decimal. Therefore, if LUN ID 10 is on a vendor that displays its IDs in hex, the LUN ID in the DM define is 16 (hex 10). An example of a hexadecimal LUN number is shown in Figure 9-49 on page 298, taken from an ESS 800. In this example you can see LUN 000E, 000F, and 0010. These are entered into the XIV data migration definitions as LUNs 14, 15, and 16, respectively. See 9.14, "Device-specific considerations" on page 304, for more details.
- The LUN ID allocated to the XIV has been allocated to an incorrect XIV WWPN. Make sure that the proper volume is allocated to the correct XIV WWPNs.
- If multiple DM targets are defined, the wrong target might have been chosen when the DM was defined.
- Sometimes when volumes are added after the initial connectivity is defined, the volume is not available. Go to the Migration Connectivity panel and delete the links between the XIV and non XIV storage system. Delete only the links. There is no need to delete anything else. After all links are deleted, re-create the links. Go back to the DM panel

and re-create the DM. See step 6 on page 258 (in “Define non XIV storage system on the XIV (as a migration target)” on page 256).

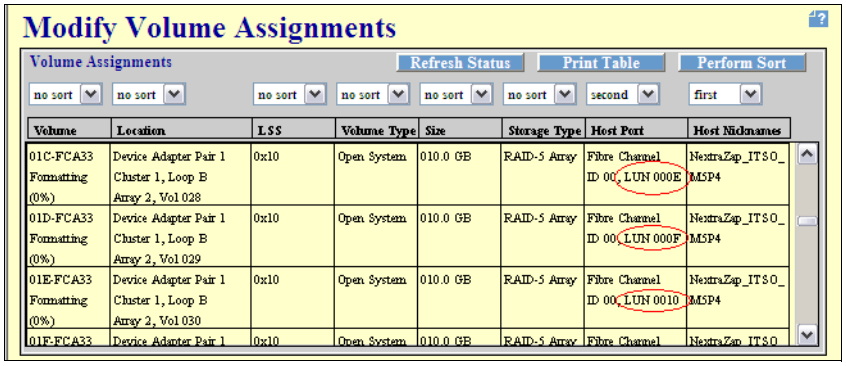


Figure 9-49 ESS 800 LUN numbers

- The volume on the source non XIV storage system might not have been initialized or low-level formatted. If the volume has data on it then this is not the case. However, if you are assigning new volumes from the non XIV storage system then perhaps these new volumes have not completed the initialization process. On ESS 800 storage the initialization process can be displayed from the Modify Volume Assignments panel. In Figure 9-49 the volumes are still 0% background-formatted, so they are not accessible by the XIV. So for ESS 800, keep clicking **Refresh Status** on the ESS 800 web GUI until the formatting message disappears.

9.11.3 Local volume is not formatted

This error occurs when a volume that already exists is chosen as the *destination name* and has already been written either from a host or a previous DM process that has since been removed from the DM panel. To work around this error do one of the following tasks:

- Use another volume as a migration destination.
- Delete the volume that you are trying to migrate to and then create it again.
- Go to the **Volumes** → **Volumes and Snapshots** panel. Right-click to select the volume and choose **Format**.

Warning: his deletes all data currently on the volume without recovery. A warning message is displayed to challenge the request.

9.11.4 Host server cannot access the XIV migration volume

This error occurs if you attempt to read the contents of a volume on a non XIV storage device through an XIV data migration without activating the data migration. This happens if the migration is performed without following the correct order of steps. The server should not attempt to access the XIV volume being migrated until the XIV shows that the migration is initializing and active (even if the progress percentage only shows 0%) or fully synchronized.

Note: This might also happen in a cluster environment where the XIV is holding a SCSI reservation. Make sure all nodes of a cluster are shutdown prior to starting a migration. The XCLI command **`reservation_list`** will list all scsi reservations held by the XIV. Should a volume be found with reservations where all nodes are offline, the reservations might be removed using the XCLI command **`reservation_clear`**. See XCLI documentation for further details.

9.11.5 Remote volume cannot be read

This error occurs when a volume is defined down the passive path on an active/passive multipathing storage device. This can occur in several cases:

- Two paths were defined on a target (non XIV storage system) that only supports active/passive multipathing. XIV is an active/active storage device. Defining two paths on any given target from an active/passive multipathing storage device is not supported. Redefine the target with only one path. Another target can be defined with one connection to the other controller. For example, if the non XIV storage system has two controllers, but the volume can only be active on one at time, controller A can be defined as one target on the XIV and controller B can be defined as a different target. In this manner, all volumes that are active on controller A can be migrated down the XIV A target and all volumes active on the controller B can be migrated down the XIV B target.
- When defining the XIV initiator to an active/passive multipathing non XIV storage system, certain storage devices allow the initiator to be defined as *not* supporting failover. The XIV initiator should be configured to the non XIV storage system in this manner. When configured as such, the volume on the passive controller is not presented to the initiator (XIV). The volume is only presented down the active controller.

See 9.3.1, “Multipathing with data migrations” on page 249 and 9.14, “Device-specific considerations” on page 304, for additional information.

9.11.6 LUN is out of range

XIV currently supports migrating data from LUNs with a LUN ID less than 512 (decimal). This is usually not an issue, as most non XIV storage systems, by default, present volumes on an initiator basis. For example, if there are three hosts connected to the same port on a non XIV storage system, each host can be allocated volumes starting at the same LUN ID. So, for migration purposes, you must either map one host at a time (and then reuse the LUN IDs for the next host) or use different sequential LUN numbers for migration. For example, if three hosts each have three LUNs mapped using LUN IDs 20, 21, and 22, for migration purposes, migrate them as follows:

1. LUN IDs 30, 31, 32 (first host)
2. LUN IDs 33, 34, 35 (second host)
3. LUN IDs 36, 37, 38 (third host)

Then, from the XIV you can again map them to each host as LUN IDs 20, 21, and 22 (as they were from the non XIV storage).

If migrating from an EMC Symmetrix or DMX there are special considerations.
See 9.14.2, “EMC Symmetrix and DMX” on page 306.

9.12 Backing out of a data migration

For change management purposes, you might be required to document a back-out procedure. Four possible places in the migration process where backing out might occur are described in this section.

9.12.1 Back out before migration is defined on the XIV

If a data migration definition does not exist yet, then no action must be taken on the XIV. You can simply zone the host server back to the non XIV storage system and unmap the host server's LUNs away from the XIV and back to the host server, taking care to ensure that the correct LUN order is preserved.

9.12.2 Back out after a data migration has been defined but not activated

If the data migration definition exists but has not been activated, then you can follow the same steps as described in 9.12.1, "Back out before migration is defined on the XIV" on page 300. To remove the inactive migration from the migration list you must delete the XIV volume that was going to receive the migrated data.

9.12.3 Back out after a data migration has been activated but is not complete

If the data migration shows in the GUI to have a status of `Initialization` or the XCLI shows it as `active=yes`, then the background copy process was started. Do *not* deactivate the migration in this state as you will block any I/O passing through the XIV from the host server to the migration LUN on the XIV and to the LUN on the non XIV disk system. You must shut down the host server or its applications first. After doing this you can deactivate the data migration and then if you want, you can delete the XIV data migration volume. Then restore the original LUN masking and SAN fabric zoning and bring your host back up.

Important: If you chose to not allow source updating and write I/O has occurred after the migration started, then the contents of the LUN on the non XIV storage system will not contain the changes from those writes. Understanding the implications of this is important in a back-out plan. The preference is to use the Keep Source Updated option.

9.12.4 Back out after a data migration has reached the synchronized state

If the data migration shows in the GUI as having a status of synchronised, then the background copy has completed. In this case back-out can still occur because the data migration is not destructive to the source LUN on the non XIV storage system. Simply reverse the process by shutting down the host server or applications and restore the original LUN masking and switch zoning settings. You might need to also reinstall the relevant host server multipath software for access to the non XIV storage system.

Important: If you chose to not allow source updating and write I/O has occurred during the migration or after it has completed, then the contents of the LUN on the non XIV storage system do not contain the changes from those writes. Understanding the implications of this is important in a back-out plan. Use the Keep Source Updated option.

9.13 Migration checklist

There are three separate stages to a migration cutover:

1. Prepare the environment for the implementation of the XIV.
2. Cut over your hosts.
3. Remove any old devices and definitions as part of a cleanup stage.

For site setup, the high-level process is as follows:

1. Install XIV and cable it into the SAN.
2. Pre-populate SAN zones in switches.
3. Pre-populate the host/cluster definitions in the XIV.
4. Define XIV to non XIV disk as a host.
5. Define non XIV disk to XIV as a migration target and confirm paths.

Then, for each host, the high-level process is as follows:

1. Update host drivers, install Host Attachment Kit, and shut down the host.
2. Disconnect/un-zone the host from non XIV storage and then zone the host to XIV.
3. Map the host LUNs away from the host instead of mapping them to the XIV.
4. Create XIV data migration (DM).
5. Map XIV DM volumes to the host.
6. Open the host.

When all data on the non XIV disk system has been migrated, perform site cleanup:

1. Delete all SAN zones related to the non XIV disk.
2. Delete all LUNs on non XIV disk and remove it from the site.

Table 9-1 shows the site setup checklist.

Table 9-1 Physical site setup checklist

Task number	Complete	Where to perform	Task
1		Site	Install XIV.
2		Site	Run fiber cables from SAN switches to XIV for host connections and migration connections.
3		Non XIV storage	Select host ports on the non XIV storage to be used for migration traffic. These ports do not have to be dedicated ports. Run new cables if necessary.
4		Fabric switches	Create switch aliases for each XIV Fibre Channel port and any new non XIV ports added to the fabric.
5		Fabric switches	Define SAN zones to connect hosts to XIV (but do not activate the zones). You can do this by cloning the existing zones from host to non XIV disk and swapping non XIV aliases for new XIV aliases.
6		Fabric switches	Define and activate SAN zones to connect non XIV storage to XIV initiator ports (unless direct connected).
7		Non XIV storage	If necessary, create a small LUN to be used as LUN0 to allocate to the XIV.
8		Non XIV storage	Define the XIV on the non XIV storage system, mapping LUN0 to test the link.
9		XIV	Define non XIV storage to the XIV as a migration target and add ports. Confirm that links are green and working.
10		XIV	Change the max_initialization_rate depending on the non XIV disk. You might want to start at a smaller value and increase it if no issues are seen.
11		XIV	Define all the host servers to the XIV (cluster first if using clustered hosts). Use a host listing from the non XIV disk to get the WWPNs for each host.
12		XIV	Create storage pools as required. Ensure that there is enough pool space for all the non XIV disk LUNs being migrated.

After the site setup is complete, the host migrations can begin. Table 9-2 shows the host migration checklist. Repeat this checklist for every host. Task numbers that are identified with a gray background (steps 5-27) must be performed with the host application offline.

Table 9-2 Host Migration to XIV checklist

Task number	Complete	Where to perform	Task
1		Host	From the host, determine the volumes to be migrated and their relevant LUN IDs and hardware serial numbers or identifiers.
2		Host	UNIX/Linux servers: Document/Save LVM configuration including PVIDs, VGIDs, and LVIDs
3		Host	If the host is remote from your location, confirm that you can power the host back on after shutting it down (using tools such as an RSA card or IBM BladeCenter manager).
4		Non XIV storage	Get the LUN IDs of the LUNs to be migrated from non XIV storage system. Convert from hex to decimal if necessary.

Task number	Complete	Where to perform	Task
5		Host	Shut down the application.
6		Host	Set the application to *not* start automatically at reboot. This helps when performing administrative functions on the server (upgrades of drivers, patches, and so on) and verifying all LUNs are present before allowing the application to start.
7		Host	UNIX/Linux server: Export LVM volume groups. Though not required, it is recommended so that when booting the server after starting the DM and allocating the XIV LUNs, one can verify all LUNs are available before starting LVM.
8		Host	UNIX/Linux servers: Comment out disk mount points on affected disks in the mount configuration file. This helps with system reboots while configuring for XIV.
9		Host	Shut down affected servers.
10		Fabric	Change the active zoneset to exclude the SAN zone that connects the host server to non XIV storage and include the SAN zone for the host server to XIV storage. The new zone should have been created during site setup.
11		Non XIV storage	Unmap source volumes from the host server.
12		Non XIV storage	Map source volumes to the XIV host definition (created during site setup).
13		XIV	Create data migration pairing (XIV volumes created dynamically).
14		XIV	Test XIV migration for each volume.
15		XIV	Start XIV migration and verify it. If you want, wait for migration to finish.
16		Host	Boot the server. (Be sure that the server is *not* attached to any storage.)
17		Host	Co-existence of non XIV and XIV multipathing software is supported with an approved SCORE(RPQ) only. Remove any unapproved multipathing software
18		Host	Install patches, update drivers, and HBA firmware as necessary.
19		Host	Install the XIV Host Attachment Kit. (Be sure to note prerequisites.)
20		Host	At this point you might need to reboot (depending on operating system).
21		XIV	Map XIV volumes to the host server. (Use original LUN IDs.)
22		Host	Scan for new volumes.
23		Host	Verify that the LUNs are available and that pathing is correct.
24		Host	UNIX/Linux servers: Import LVM volume groups *if* exported in task 2.
25		Host	UNIX/Linux Servers: Update mount points for new disks in the mount configuration file if they have changed. Mount the file systems.
26		Host	Start the application.
27		Host	Set the application to start automatically if this was previously changed.
28		XIV	Monitor the migration if it is not already completed.

Task number	Complete	Where to perform	Task
29		XIV	When the volume is synchronized delete the data migration (do *not* deactivate the migration).
30		Non XIV Storage	Unmap migration volumes away from XIV if you must free up LUN IDs.
31		XIV	Consider re-sizing the migrated volumes to the next 17 GB boundary if the host operating system is able to use new space on a resized volume.
32		Host	If XIV volume was resized, use host procedures to use the extra space.
33		Host	If non XIV storage system drivers and other supporting software were not removed earlier, remove them when convenient.

When all the hosts and volumes have been migrated, several site cleanup tasks remain, as shown in Table 9-3.

Table 9-3 Site cleanup checklist

Task number	Complete	Where to perform	Task
1		XIV	Delete migration paths and targets.
2		Fabric	Delete all zones related to non XIV storage including the zone for XIV migration.
3		Non XIV storage	Delete all LUNs and perform secure data destruction if required.

9.14 Device-specific considerations

The XIV supports migration from practically any SCSI storage device that has Fibre Channel interfaces. This section contains device-specific information, but the list is not exhaustive. Ensure that you understand the following requirements for your storage device:

LUN0 Do you need to specifically map a LUN to LUN ID zero? This question determines whether you will have a problem defining the paths.

LUN numbering Does the storage device GUI or CLI use decimal or hexadecimal LUN numbering? This question determines whether you must do a conversion when entering LUN numbers into the XIV GUI.

Multipathing Is the device active/active or active/passive? This question determines whether you define the storage device as a single target or as one target per internal controller or service processor.

Definitions Does the device have specific requirements when defining hosts?

Converting hexadecimal LUN IDs to decimal LUN IDs

When mapping volumes to the XIV be sure to note the LUN IDs allocated by the non XIV storage. The methodology to do this varies by vendor and device. If the device uses hexadecimal LUN numbering then be sure you understand how to convert hexadecimal numbers into decimal numbers, to enter into the XIV GUI.

Using a spreadsheet to convert hex to decimal

Microsoft Excel and Open Office both have a spreadsheet formula known as hex2dec. If, for example, you enter a hexadecimal value into spreadsheet cell location A4, then the formula to convert the contents of that cell to decimal is `=hex2dec(A4)`. If this formula does not seem to work in Excel, add the Analysis ToolPak (within Excel, select **Tools** → **Add ins**, and then select **Analysis ToolPak**).

Using Microsoft calculator to convert hex to decimal

Start the calculator with the following steps:

1. Selecting **Program Files** → **Programs** → **Accessories** → **Calculator**.
2. From the View drop-down menu change from Standard to **Scientific**.
3. Select **Hex**.
4. Enter a hexadecimal number and then select **Dec**.

The hexadecimal number is converted to decimal.

Given that the XIV supports migration from almost any storage device, it is impossible to list the methodology to get LUN IDs from each one.

9.14.1 EMC CLARiiON

The following considerations are identified specifically for EMC CLARiiON:

- LUN0

  There is no requirement to map a LUN to LUN ID 0 for the CLARiiON to communicate with the XIV.

- LUN numbering

  The EMC CLARiiON uses decimal LUN numbers for both the CLARiiON ID and the host ID (LUN number).

- Multipathing

  The EMC CLARiiON is an active/passive storage device. Therefore each storage processor (SP-A and SP-B) must be defined as a separate target to the XIV. You might choose to move LUN ownership of all the LUNs that you are migrating to a specific SP and simply define only that SP as a target. But the preference is to define separate XIV targets for each SP. Moving a LUN from one SP to another is known as trespassing.

Notes:

- Several newer CLARiiON devices (CX3, CX4) use ALUA when presenting LUNS to the host and therefore appear to be an active/active storage device. ALUA is effectively masking which SP owns a LUN on the back end of the CLARiiON. Though this appears as an active/active storage device, ALUA could cause performance issues with XIV migrations if configured using active/active storage device best practices (that is, two paths for each target). This is because LUN ownership could be switching from one SP to another in succession during the migration, with each switch taking processor and I/O cycles.
- You can configure two paths to the SAME SP to two different XIV interface modules for some redundancy. This will not protect against a trespass, but might protect from an XIV hardware or SAN path failure.

- Requirements when defining the XIV

  If migrating from an EMC CLARiiON use the settings shown in Table 9-4 to define the XIV to the CLARiiON. Ensure that Auto-trespass is disabled for every XIV initiator port (WWPN) registered to the Clariion.

Table 9-4 Defining an XIV to the EMC CLARiiON

Initiator information	Setting
Initiator type	CLARiiON Open
HBA type	Host
Array CommPath	Enabled
Failover mode	0
Unit serial number	Array

9.14.2 EMC Symmetrix and DMX

Considerations in this section are identified specifically for EMC Symmetrix and DMX.

LUN0

A requirement exists for the EMC Symmetrix or DMX to present a LUN ID 0 to the XIV so that the XIV Storage System can communicate with the EMC Symmetrix or DMX. In many installations, the VCM device is allocated to LUN-0 on all FAs and is automatically presented to all hosts. In these cases, the XIV connects to the DMX with no issues. However, in newer installations, the VCM device is no longer presented to all hosts and therefore a real LUN-0 is required to be presented to the XIV so that the XIV can connect to the DMX. This LUN-0 can be a dummy device of any size that will not be migrated or an actual device that will be migrated.

LUN numbering

The EMC Symmetrix and DMX, by default, does not present volumes in the range of 0 to 511 decimal. The Symmetrix/DMX presents volumes based on the LUN ID that was given to the volume when the volume was placed on the FA port. If a volume was placed on the FA with a LUN ID of 90, this is how it is presented to the host by default. The Symmetrix/DMX also presents the LUN IDs in hex. Thus, LUN ID 201 equates to decimal 513, which is greater than 511 and is outside of the XIV's range. There are two disciplines for migrating data from a Symmetrix/DMX where the LUN ID is greater than 511 (decimal).

Remap the volume

One way to migrate a volume with a LUN ID higher than 511 is to re-map the volume in one of two ways:

- Map the volume to a free FA or an FA that has available LUN ID slots less than hex 0x200 (decimal 512). In most cases this can be done without interruption to the production server. The XIV is zoned and the target defined to the FA port with the lower LUN ID.
- Re-map the volume to a lower LUN ID, one that is less than 200 hex. However, this requires that the host be shut down while the change is taking place and is therefore not the best option.

LUN-Offset

With EMC Symmetrix Enginuity code 68 - 71, there is an EMC method of presenting LUN IDs to hosts other than the LUN ID given to the volume when placed on the FA. In Symmetrix/DMX, a volume is given a unique LUN ID when configured on an FA. Each volume on an FA must have a unique LUN ID. The default method (and a best practice of presenting volumes to a host) is to use the LUN ID given to the volume when placed on the FA. This means that if vol1 was placed on an FA with an ID of 7A (hex 0x07a, decimal 122), this is the LUN ID that is presented to the host. Using the **`lunoffset`** option of the **`symmask`** command, a volume can be presented to a host (WWPN initiator) with a different LUN ID than was assigned to the volume when placed on the FA. Because it is done at the initiator level, the production server can keep the high LUNs (above 128) while being allocated to the XIV using lower LUN IDs (below 512 decimal).

Migrating volumes that were used by HP-UX

For HP-UX hosts attached to EMC Symmetrix there is a setting known as Volume_Set_Addressing that can be enabled on a per-FA basis. This is required for HP-UX host connectivity but is *not* compatible with any other host types (including XIV). If Volume_Set_Addressing (also referred to as the *V* bit setting) is enabled on an FA, then the XIV will not be able to access anything but LUN 0 on that FA. To avoid this issue, map the HP-UX host volumes to a different FA that is not configured specifically for HP-UX. Then zone the XIV migration port to this FA instead of the FA being used by HP-UX. in most cases, EMC symmetrix/DMX volumes can be mapped to an additional FA without interruption.

Multipathing

The EMC Symmetrix and DMX are active/active storage devices.

9.14.3 HDS TagmaStore USP

This section describes LUN0, numbering, and multipathing for HDS TagmaStore USP.

LUN0

There is a requirement for the HDS TagmaStore Universal Storage Platform (USP) to present a LUN ID 0 to the XIV so that the XIV Storage System can communicate with the HDS device.

LUN numbering

The HDS USP uses hexadecimal LUN numbers.

Multipathing

The HDS USP is an active/active storage device.

9.14.4 HP EVA

The following requirements were determined after migration from a HP EVA 4400 and 8400.

LUN0

There is no requirement to map a LUN to LUN ID 0 for the HP EVA to communicate with the XIV. This is because by default the HP EVA presents a special LUN known as the Console LUN as LUN ID 0.

LUN numbering

The HP EVA uses decimal LUN numbers.

Multipathing

The HP EVA 4000/6000/8000 are active/active storage devices. For HP EVA 3000/5000, the initial firmware release was active/passive, but a firmware upgrade to VCS Version 4.004 made it active/active capable. For more details, see the following website:

http://h21007.www2.hp.com/portal/site/dspp/menuitem.863c3e4cbcdc3f3515b49c108973a801?ciid=aa08d8a0b5f02110d8a0b5f02110275d6e10RCRD

Requirements when connecting to XIV

Define the XIV as a Linux host.

To check the LUN IDs assigned to a specific host, use the following steps:

1. Log in into Command View EVA.h.
2. Select the storage on which you are working.
3. Click the Hosts icon.
4. Select the specific host.
5. Click the **Presentation** tab.
6. Here you will see the LUN name and the LUN ID presented.

To present EVA LUNs to XIV, use the following steps:

1. Create the host alias for XIV and add the XIV initiator ports that are zoned to EVA.
2. From the Command View EVA, select the active VDisk that must be presented to XIV.
3. Click the **Presentation** tab.
4. Click **Present**.
5. Select the XIV host Alias created.
6. Click the **Assign** LUN button on top.
7. Specify the LUN ID that you want to specify for XIV. Usually this is the same as was presented to the host when it was accessing the EVA.

9.14.5 IBM DS3000, DS4000, DS5000

The following considerations were identified specifically for DS4000 but apply for all models of D3000, DS4000, and DS5000 (for purposes of migration they are functionally all the same). For ease of reading, only the DS4000 is referenced.

LUN0

There is a requirement for the DS4000 to present a LUN on LUN ID 0 to the XIV to allow the XIV to communicate with the DS4000. It might be easier to create a new 1 GB LUN on the DS4000 just to satisfy this requirement. This LUN does not need to have any data on it.

LUN numbering

For all DS4000 models, the LUN ID used in mapping is a *decimal* value between 0 to 15 or 0 to 255 (depending on model). This means that no hex-to-decimal conversion is necessary. Figure 9-13 on page 262 shows an example of how to display the LUN IDs.

Defining the DS4000 to the XIV as a target

The DS4000 is an active/passive storage device. This means that each controller on the DS4000 must be defined as a separate target to the XIV. You must take note of which volumes are currently using which controllers as the active controller.

Preferred path errors

The following issues can occur if you have misconfigured a migration from a DS4000. You might initially notice that the progress of the migration is slow. The DS4000 event log might contain errors, such as the one shown in Figure 9-50. If you see the migration volume fail between the A and B controllers, this means that the XIV is defined to the DS4000 as a host that supports ADT/RDAC (which you should immediately correct) and that either the XIV target definitions have paths to both controllers or that you are migrating from the wrong controller.

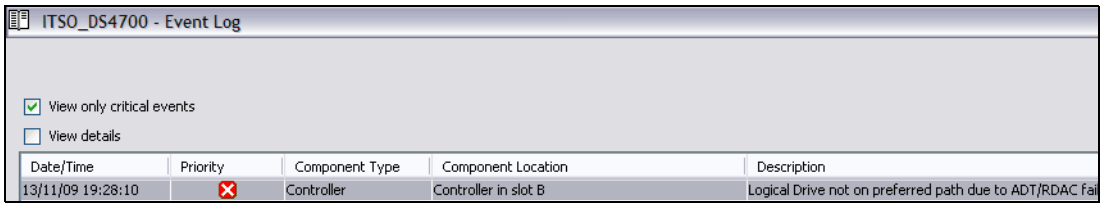


Figure 9-50 DS4000 LUN failover

In Example 9-10 the XCLI commands show that the target called ITSO_DS4700 has two ports, one from controller A (201800A0B82647EA) and one from controller B (201900A0B82647EA). This is not the correct configuration and should not be used.

Example 9-10 Incorrect definition, as target has ports to both controllers

```
>> target_list
Name                SCSI Type   Connected
ITSO_DS4700           FC          yes

>> target_port_list target=ITSO_DS4700
Target Name          Port Type   Active   WWPN              iSCSI Address   iSCSI Port
ITSO_DS4700          FC          yes      201800A0B82647EA                  0
ITSO_DS4700          FC          yes      201900A0B82647EA                  0
```

Instead, two targets should have been defined, as shown in Example 9-11. In this example, two separate targets have been defined, each target having only one port for the relevant controller.

Example 9-11 Correct definitions for a DS4700

```
> target_list
Name                SCSI Type   Connected
DS4700-ctrl-A       FC          yes
DS4700-ctrl-B       FC          yes

>> target_port_list target=DS4700-Ctrl-A
Target Name          Port Type   Active   WWPN              iSCSI Address   iSCSI Port
DS4700-ctrl-A        FC          yes      201800A0B82647EA                  0

>> target_port_list target=DS4700-Ctrl-B
Target Name          Port Type   Active   WWPN              iSCSI Address   iSCSI Port
DS4700-ctrl-B        FC          yes      201900A0B82647EA                  0
```

Note: Although some of the DS4000 storage devices (for example, DS4700) have multiple target ports on each controller, it will not help you to attach more target ports from the same controller because XIV does not have multipathing capabilities. Only one path per controller should be attached.

Defining the XIV to the DS4000 as a host

Use the DS Storage Manager to check the profile of the DS4000 and select a host type for which ADT is disabled or failover mode is RDAC. To display the profile from the DS Storage Manager choose **Storage Subsystem** → **View** → **Profile** → **All**. Then go to the bottom of the Profile panel. The profile might vary according to NVRAM version. In Example 9-12 select the host type for which ADT status is disabled (Windows 2000).

Example 9-12 Earlier NVRAM versions

```
HOST TYPE                                          ADT STATUS
Linux                                              Enabled
Windows 2000/Server 2003/Server 2008 Non-Clustered Disabled
```

In Example 9-13 choose the host type that specifies RDAC (Windows 2000).

Example 9-13 Later NVRAM versions

```
HOST TYPE                                          FAILOVER MODE
Linux                                              ADT
Windows 2000/Server 2003/Server 2008 Non-Clustered RDAC
```

You can now create a host definition on the DS4000 for the XIV. If you have zoned the XIV to both DS4000 controllers you can add both XIV initiator ports to the host definition. This means that the host properties should look similar to Figure 9-51. After mapping your volumes to the XIV migration host, you must take note of which controller each volume is owned by. When you define the data migrations on the XIV, the migration should point to the target that matches the controller that owns the volume being migrated.

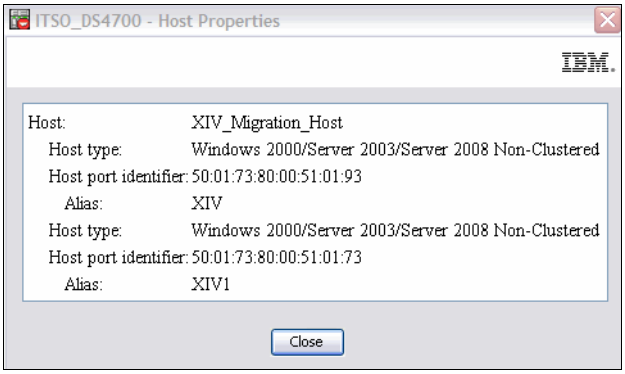


Figure 9-51 XIV defined to the DS4000 as a host

9.14.6 IBM ESS 800

The following considerations were identified for IBM ESS 800.

LUN0

There is no requirement to map a LUN to LUN ID 0 for the ESS to communicate with the XIV.

LUN numbering

The LUN IDs used by the ESS are in *hexadecimal*, so they must be converted to decimal when entered as XIV data migrations. It is not possible to specifically request certain LUN IDs. In Example 9-14 there are 18 LUNs allocated by an ESS 800 to an XIV host called NextraZap_ITSO_M5P4. You can clearly see that the LUN IDs are hex. The LUN IDs in the right column were added to the output to show the hex-to-decimal conversion needed for use with XIV. An example of how to view LUN IDs using the ESS 800 web GUI is shown in Figure 9-49 on page 298.

Restriction: The ESS can allocate LUN IDs only in the range 0 - 255 (hex 00 to FF). This means that only 256 LUNs can be migrated at one time on a per-target basis; more than 256 LUNs can be migrated if more than one target is used.

Example 9-14 Listing ESS 800 LUN IDs using ESSCLI

```
C:\esscli -s 10.10.1.10 -u storwatch -p specialist list volumeaccess -d
"host=NextraZap_ITSO_M5P4"
Tue Nov 03 07:20:36 EST 2009 IBM ESSCLI 2.4.0

Volume  LUN   Size(GB)  Initiator         Host
------  ----  --------  ----------------  ------------------
 100e   0000      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 0)
 100f   0001      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 1)
 1010   0002      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 2)
 1011   0003      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 3)
 1012   0004      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 4)
 1013   0005      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 5)
 1014   0006      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 6)
 1015   0007      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 7)
 1016   0008      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 8)
 1017   0009      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 9)
 1018   000a      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 10)
 1019   000b      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 11)
 101a   000c      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 12)
 101b   000d      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 13)
 101c   000e      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 14)
 101d   000f      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 15)
 101e   0010      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 16)
 101f   0011      10.0  5001738000230153  NextraZap_ITSO_M5P4 (LUN ID is 17)
```

Multipathing

The ESS 800 is an active/active storage device. You can define multiple paths from the XIV to the ESS 800 for migration. Ideally, connect to more than one host bay in the ESS 800. Because each XIV host port is defined as a separate host system, ensure that the LUN ID used for each volume is the same. The Modify Volume Assignments panel has the `Use same ID/LUN in source and target` check box that can assist you. Figure 9-69 on page 325 shows a good example of two XIV host ports with the same LUN IDs.

Requirements when defining the XIV

Define each XIV host port to the ESS 800 as a *Linux x86* host.

9.14.7 IBM DS6000 and DS8000

The following considerations were identified for DS6000 and DS8000.

LUN0

There is no requirement to map a LUN to LUN ID 0 for a DS6000 or DS8000 to communicate with the XIV.

LUN numbering

The DS6000 and DS8000 use hexadecimal LUN IDs. These can be displayed using DSCLI with the **showvolgrp -lunmap *xxx*** command, where *xxx* is the volume group created to assign volumes to the XIV for data migration. Do not use the web GUI to display LUN IDs.

Multipathing with DS6000

The DS6000 is an active/active storage device, but each controller has dedicated host ports, whereas each LUN has a preferred controller. If I/O for a particular LUN is sent to host ports of the non-preferred controller, the LUN will not fail over, but that I/O might experience a small performance penalty. This might lead you to consider migrating volumes with *even* LSS numbers (such as volumes 0000 and 0200) from the upper controller and volumes with *odd* LSS numbers (such as volumes 0100 and 0300) from the lower controller. However, this is not a robust solution. Define the DS6000 as a single target with one path to each controller.

Multipathing with DS8000

The DS8000 is an active/active storage device. You can define multiple paths from the XIV to the DS8000 for migration. Ideally, connect to more than one I/O bay in the DS8000.

Requirements when defining the XIV

In Example 9-15 a volume group is created used a type of *SCSI Map 256*, which is the correct type for a Red Hat Linux host type. A starting LUN ID of 8 is chosen to show how hexadecimal numbering is used. The range of valid LUN IDs for this volume group are 0 to FF (0 to 255 in decimal). An extra LUN is then added to the volume group to show how specific LUN IDs can be selected by volume. Two host connections are then created using the Red Hat Linux host type. Using the same volume group ID for both connections ensures that the LUN numbering used by each defined path will be the same.

Example 9-15 Listing DS6000 and DS8000 LUN IDs

```
dscli> mkvolgrp -type scsimap256 -volume 0200-0204 -LUN 8 migrVG
CMUC00030I mkvolgrp: Volume group V18 successfully created.
dscli> chvolgrp -action add -volume 0205 -lun 0E V18
CMUC00031I chvolgrp: Volume group V18 successfully modified.
dscli> showvolgrp -lunmap V18
Name migrVG
ID   V18
Type SCSI Map 256
Vols 0200 0201 0202 0203 0204 0205
==============LUN Mapping===============
vol  lun
========
0200 08      (comment: use decimal value 08 in XIV GUI)
0201 09      (comment: use decimal value 09 in XIV GUI)
0202 0A      (comment: use decimal value 10 in XIV GUI)
0203 0B      (comment: use decimal value 11 in XIV GUI)
0204 0C      (comment: use decimal value 12 in XIV GUI)
```

```
-    0D
0205 0E      (comment: use decimal value 14 in XIV GUI)
dscli> mkhostconnect -wwname 5001738000230153 -hosttype LinuxRHEL -volgrp V18 XIV_M5P4
CMUC00012I mkhostconnect: Host connection 0020 successfully created.
dscli> mkhostconnect -wwname 5001738000230173 -hosttype LinuxRHEL -volgrp V18 XIV_M7P4
CMUC00012I mkhostconnect: Host connection 0021 successfully created.
dscli> lshostconnect
Name                ID   WWPN             HostType  Profile                portgrp volgrpID
==========================================================================================
XIV_M5P4            0020 5001738000230153 LinuxRHEL Intel - Linux RHEL          0 V18
XIV_M7P4            0021 5001738000230173 LinuxRHEL Intel - Linux RHEL          0 V18
```

9.14.8 IBM Storwize V7000 and SAN Volume Controller

This discussion presumes that you have an existing Storwize V7000 or SAN Volume Controller and are replacing it with an XIV. You could use the migration function built into the Storwize V7000 or SAN Volume Controller. If you choose to do this, consult *IBM XIV Gen3 with IBM System Storage SAN Volume Controller and Storwize V7000*, REDP-5063.

LUN0

There is no requirement to map a volume to LUN ID 0 for a Storwize V7000 or SAN Volume Controller to communicate with the XIV.

LUN numbering

The Storwize V7000 and SAN Volume Controller use decimal LUN IDs. These can be displayed and set using the GUI or CLI.

Multipathing with Storwize V7000 and SAN Volume Controller

The Storwize V7000 and SAN Volume Controller are active/active storage devices, but each controller has dedicated host ports and each volume or LUN has a preferred controller. If I/O for a particular LUN is sent to host ports of the non-preferred controller, the LUN will not fail over. Either controller can handle I/O for any volume. This allows you to migrate using paths from either controller using multipathing.

Requirements when connecting to XIV

There are no special requirements when defining the XIV to Storwize V7000 or SAN Volume Controller as a host.

9.14.9 N series and iSCSI setup

This section discusses N series (or NetApp) and iSCSI setup for data migration.

iSCSI target creation

Setting up the iSCSI target is almost the same as that for fiber, but more information is needed. Follow these steps to create the iSCSI target:

1. In the XIV GUI go to **Migration Connectivity** and click **Create Target**.
2. In the Define Target dialog box, create a name for the target and choose iSCSI for the target protocol (Figure 9-52 on page 314).

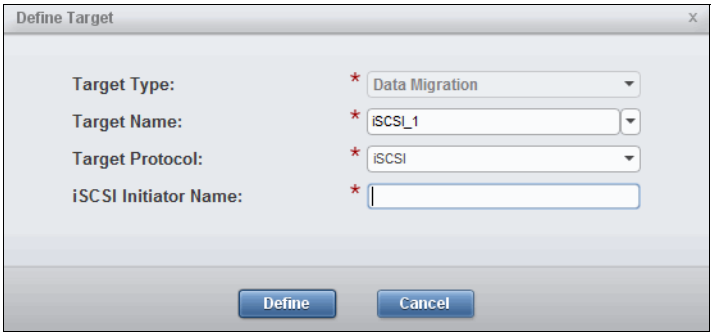


Figure 9-52 iSCSI target

3. iSCSI Initiator Name is required to set up the connection. The N series should have this information stored. In the FilerView select **LUNs** → **iSCSI** → **Reports**. The Initiator Name is stored under iSCSI Nodename in the report. Copy and paste the node name into the iSCSI Initiator Name field and click **Define**.
4. Add a port. Having previously added a Fibre Channel WWPN, you now need to enter the IP for the NetApp storage. Right-click the target system in the Migration Connectivity window and select **Add port**. In the Add Port dialog box enter the Filers IP address in the IP Address field and click **Add**.
5. Click and drag a line from an available IP port on the XIV to the new iSCSI port. If everything was set up correctly, the line connecting the two storage arrays should turn green.
6. The connection can be confirmed on the filer by selecting **LUNs** → **iSCSI** → **Initiators**. The XIVs initiator node name should be displayed here. The connection does not require a LUN presented from the N series to the XIV array to be established.

After the link is connected and operational, the XIV must be set up on the Filer as an Initiator Group and the LUNs that are going to be migrated must be added to this group.

Important: Be sure `max_initialization_rate` is set to 10. Setting the link any higher can cause the connection to go offline and then online. The following example shows how to set the link speed with XCLI. More information about this setting is in 9.7.1, “Changing the synchronization rate” on page 276.

```
target_config_sync_rates target="Netapp" max_initialization_rate=10
```

9.15 Host-specific considerations

The XIV supports migration for practically any host that has Fibre Channel interfaces. This section details some operating system considerations.

9.15.1 VMware ESX

There are several unique considerations when migrating VMware data onto an XIV system.

Using Storage vMotion

Built within VMware is a powerful migration tool commonly referred to as *Storage vMotion.* Its use is the preferred method to migrate data to XIV in VMware environments. VMware refers to migration as the process of moving a virtual machine from one host or data store to another.

Within VMware, you have the following migration options to move disks:

- Cold migration: This refers to moving a server that is off to a new host. This method could also be used to migrate the servers' storage to another host.
- Migrating a Suspended Virtual Machine: This can be used to move a suspended virtual machine to a new host. You can also use this to relocate the configuration and disk files to a new storage location.
- Migration with Storage vMotion: This is used to move the virtual disks or configuration file of a powered-on virtual machine to a new data store. This is done online and does not cause any outages to the virtual machine.

Data migration using Storage vMotion

This section shows the steps to perform data migration with XIV and Storage vMotion.

Using vMotion to do data migration to XIV is the preferred method to migrate data. Although the XIV data migration tool can be used, it requires all virtual machines to be powered off.

Important: If the Virtual Machines were set up to use raw device mappings, then you must use XIV data migration to move the data.

Complete the following steps:

1. Create a LUN on XIV and map it to the VMware server. Be sure to record the LUN number (Figure 9-53).

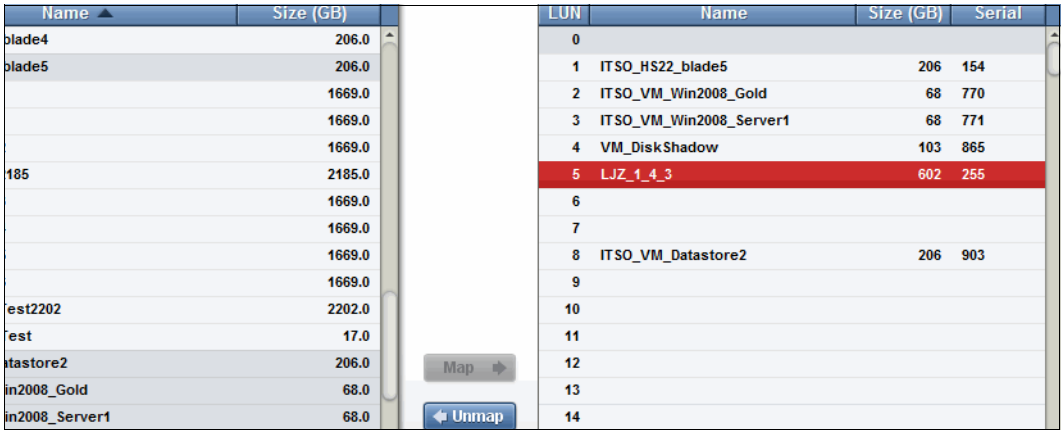


Figure 9-53 Map the LUN in XIV

2. In the vSphere client, scan for the new XIV LUN. The LUN number should match the one used to map the drive to the VMware server (Figure 9-54 on page 316).

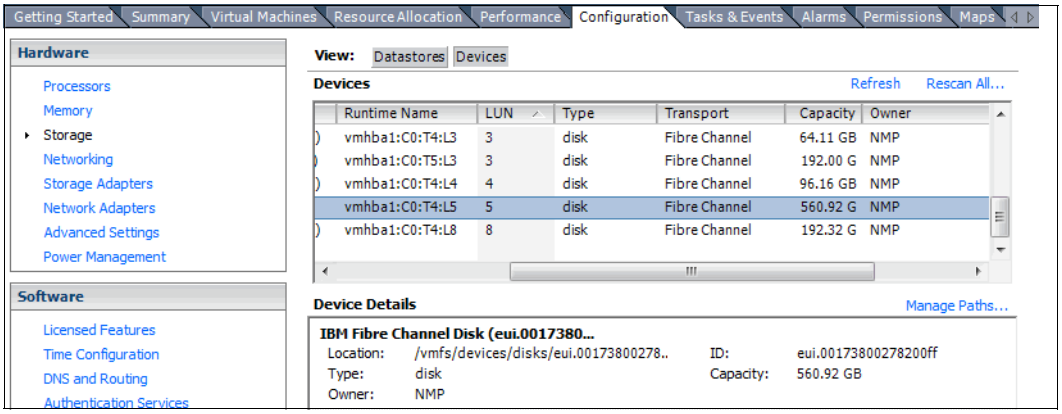


Figure 9-54 Scan for the new XIV LUN

3. Go to **Datastores** and select **Add Storage**. The Add Storage window opens (Figure 9-55).

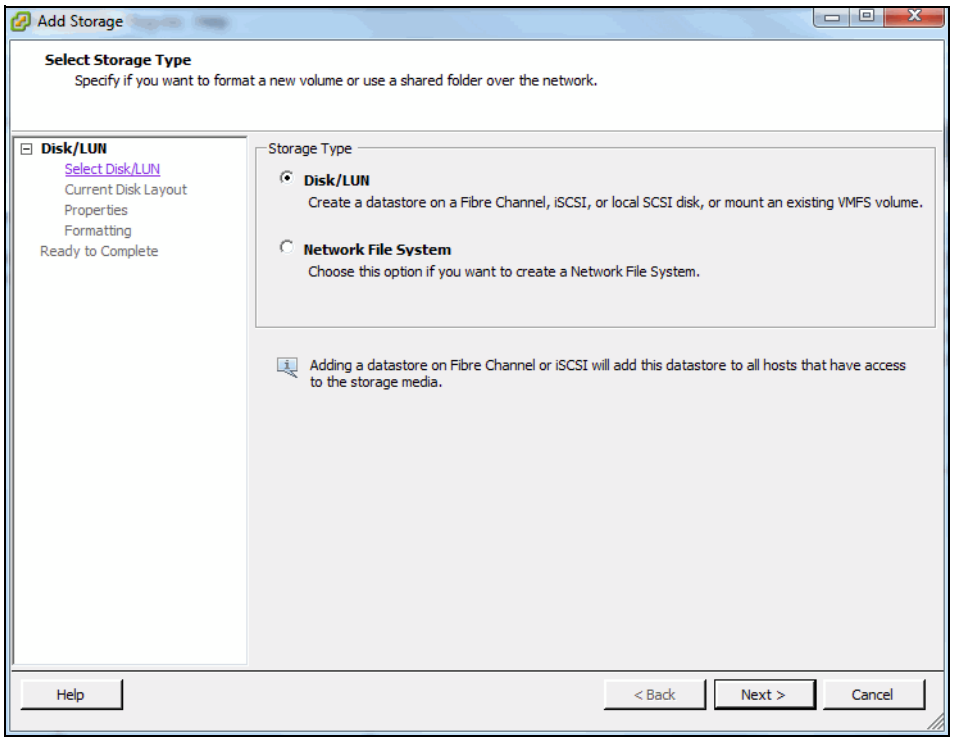

Add Storage
Select Storage Type
Specify if you want to format a new volume or use a shared folder over the network.
Disk/LUN
Select Disk/LUN
Current Disk Layout
Properties
Formatting
Ready to Complete
Storage Type
Disk/LUN
Create a datastore on a Fibre Channel, iSCSI, or local SCSI disk, or mount an existing VMFS volume.
Network File System
Choose this option if you want to create a Network File System.
Adding a datastore on Fibre Channel or iSCSI will add this datastore to all hosts that have access to the storage media.
Help
< Back
Next >
Cancel

Figure 9-55 Add storage

4. Choose the appropriate options and click **Next**. The new data store is displayed in the list (Figure 9-56 on page 317).

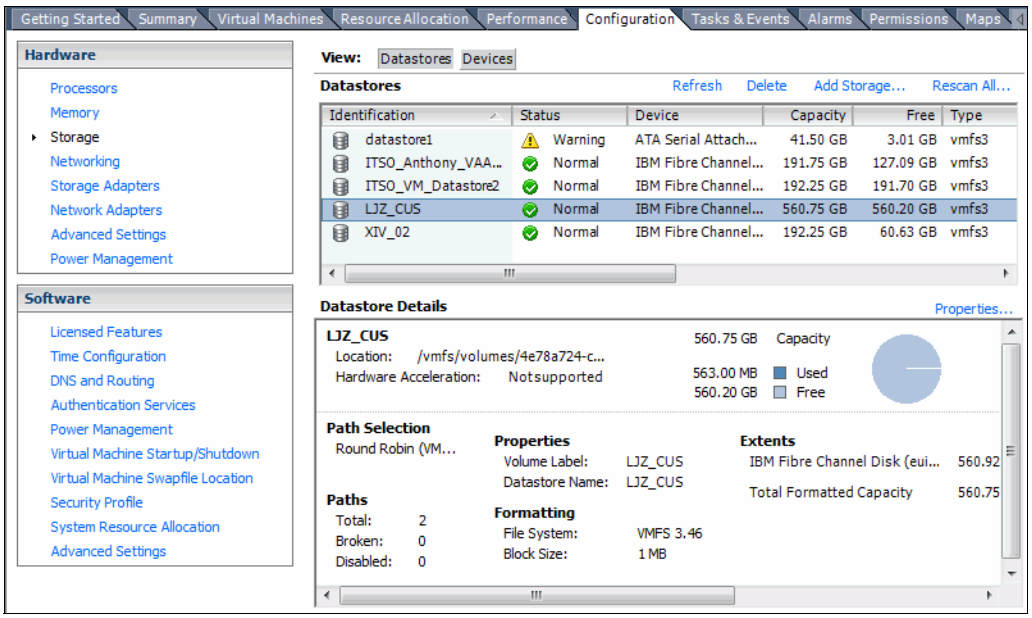


Figure 9-56 New data store

5. After the data store is created the migration can begin. Select the virtual machine that you want to migrate to a new data store. On the Summary tab, note the data store that is currently being using. Confirm that it is not a raw device. (Raw devices cannot be migrated this way; they require an XIV migration.) Right-click the virtual machine and select **Migrate** (Figure 9-57).

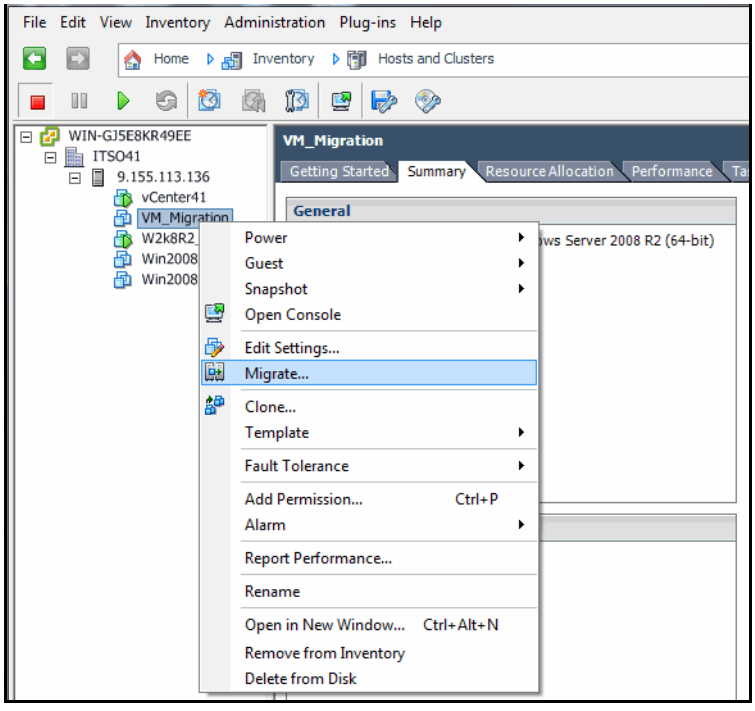


Figure 9-57 Migrate

6. Select **Change datastore** and click **Next** (Figure 9-58 on page 318).

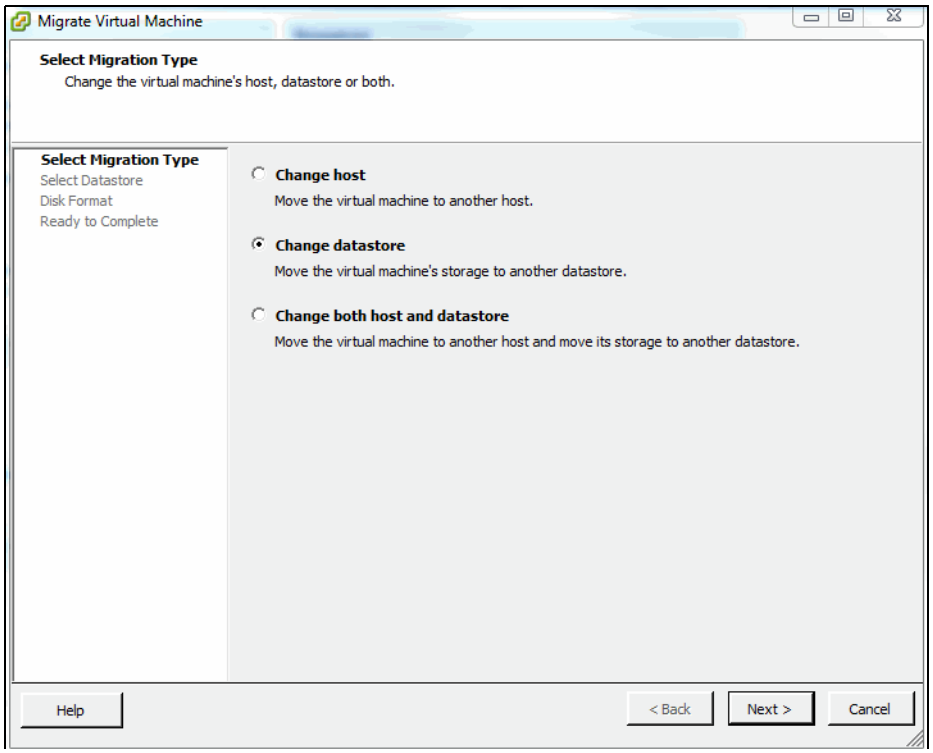

Migrate Virtual Machine
Select Migration Type
Change the virtual machine's host, datastore or both.
Select Migration Type
Select Datastore
Disk Format
Ready to Complete
Change host
Move the virtual machine to another host.
Change datastore
Move the virtual machine's storage to another datastore.
Change both host and datastore
Move the virtual machine to another host and move its storage to another datastore.
Help
< Back
Next >
Cancel

Figure 9-58 Change Datastore

7. Select the new destination for the virtual machine. The data store you created in step 4 should be available in the list (Figure 9-59).

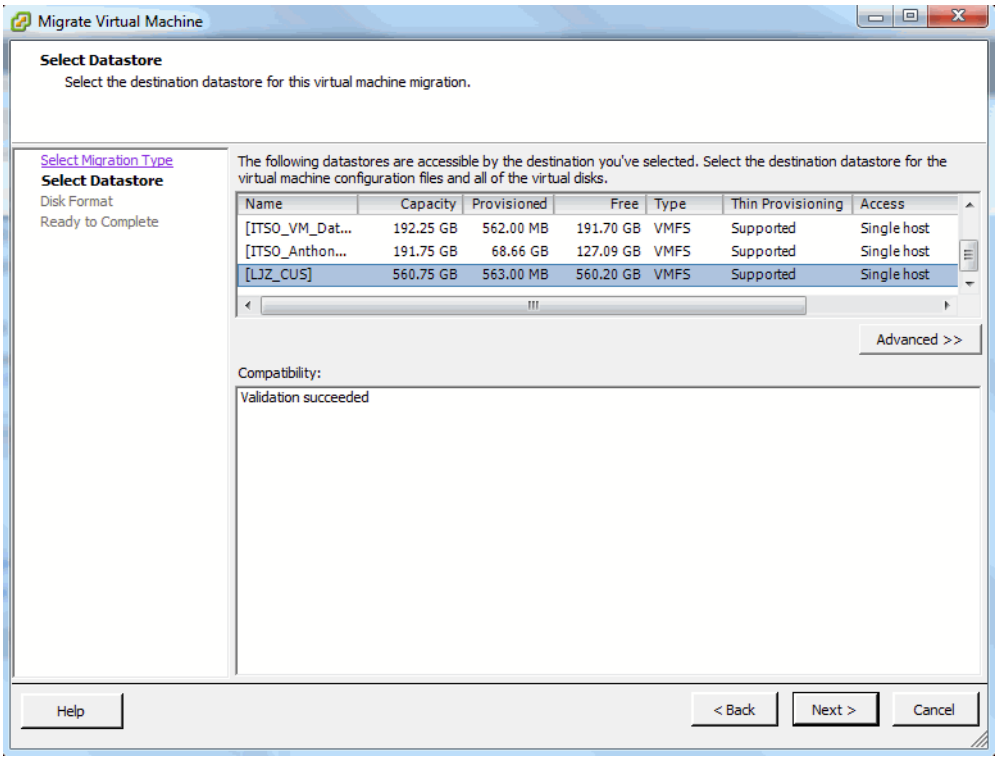


Figure 9-59 Select Datastore

8. On the next panel specify the format to use for storing the virtual disks. Although the default is `Same format as source`, you may change the format to either thick or thin (Figure 9-60).

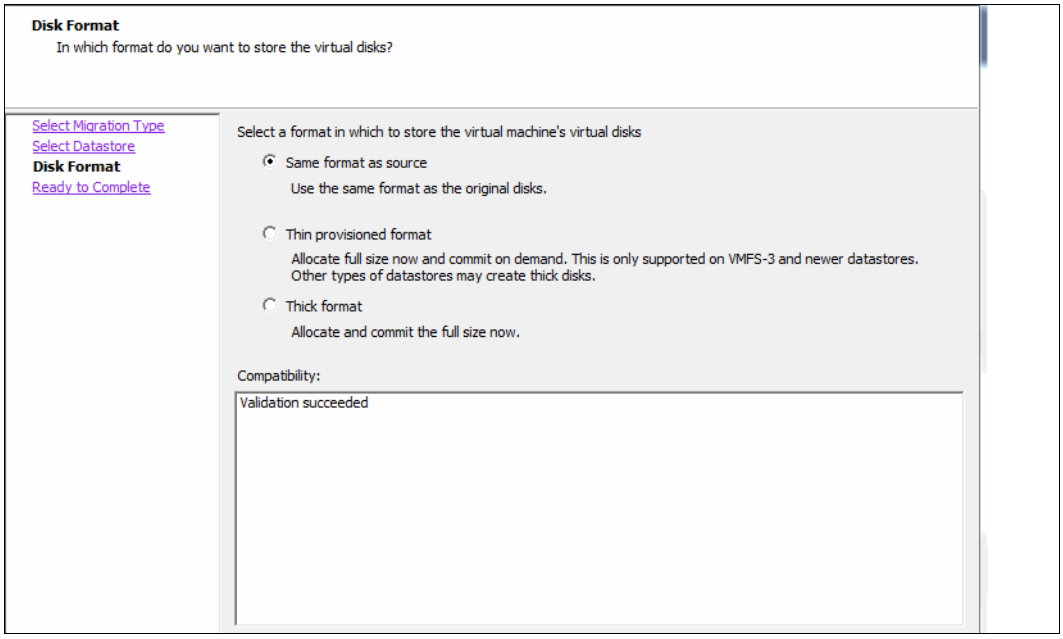

Disk Format
In which format do you want to store the virtual disks?
Select Migration Type
Select Datastore
Disk Format
Ready to Complete
Select a format in which to store the virtual machine's virtual disks
Same format as source
Use the same format as the original disks.
Thin provisioned format
Allocate full size now and commit on demand. This is only supported on VMFS-3 and newer datastores.
Other types of datastores may create thick disks.
Thick format
Allocate and commit the full size now.
Compatibility:
Validation succeeded

Figure 9-60 Choose disk format

9. The last panel displays the settings that were chosen in the migration wizard. Review the settings to confirm that the correct choices were made and click **Finish** (Figure 9-61).

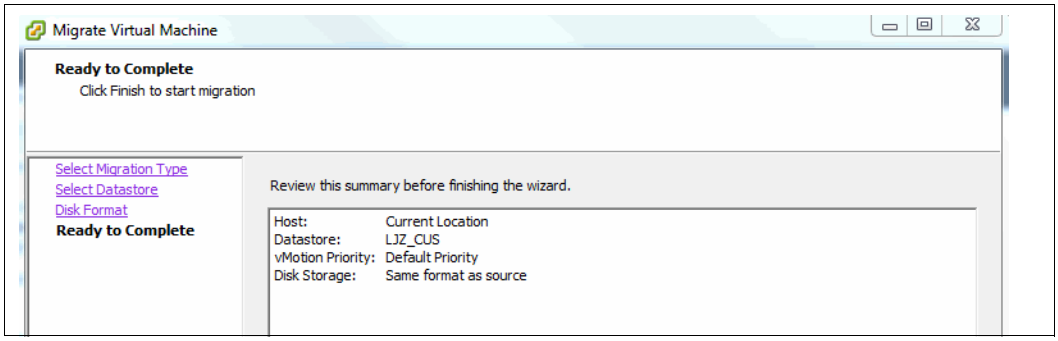


Figure 9-61 Summary

10. Migration status is visible in the Recent Tasks panel as shown in Figure 9-62.

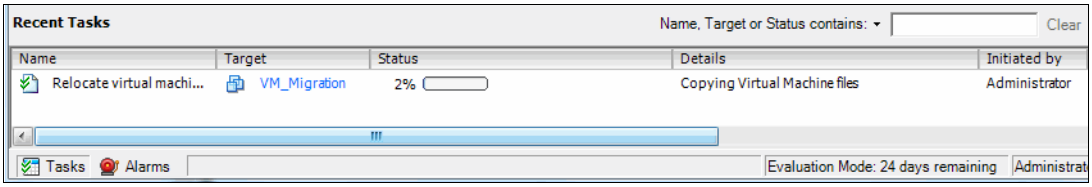


Figure 9-62 Migration status

11. After migration is finished, the status changes to Complete and the Virtual Machine now uses the new data store (Figure 9-63).

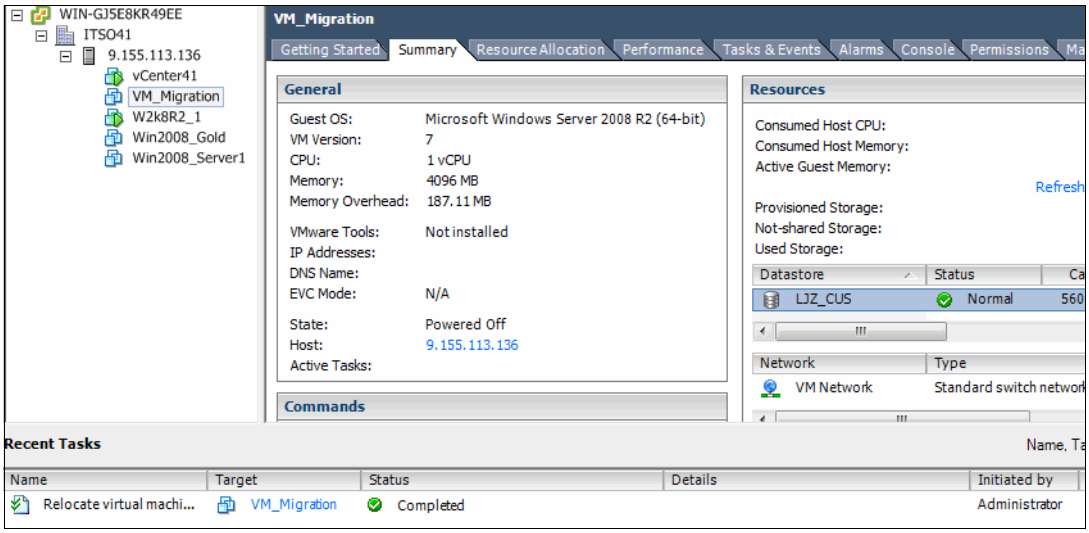


Figure 9-63 New data store

With vMotion, you can complete the entire migration process on line and the server never had to experience an outage.

Moving a raw device mapping (RDM) using XIV Data Migration

This section describes how to migrate a raw device mapping. This is effectively just a regular XIV Data Migration with some extra host-specific steps. This means that an outage is still required on the guest operating system.

Follow these steps to migrate a raw device mapping:

1. Either shut down the virtual machine or take steps within the operating system running in the virtual machine to take the disk that is mapped as a raw device off line. For instance, if the guest operating system is Windows 2008, you could stop the application using the disk and then use the Disk Management tab in Server Manager to take the disk off line.
2. Using the vSphere Client, right-click the virtual machine and select **Edit Settings.** Highlight the hard disk that is being used as a mapped raw LUN. If you are not sure which disk it is, select **Manage Paths** and take note of the Runtime Name, such as `vmhba2:C0:T0:L1`. Then from the Virtual Machine Properties tab, select the option to **Remove the Mapped Raw LUN**. Leave the default option `Remove from virtual machine` selected and click **OK**. The name of the hard disk will now have a strike-out line through it, as shown in Figure 9-64 on page 321.

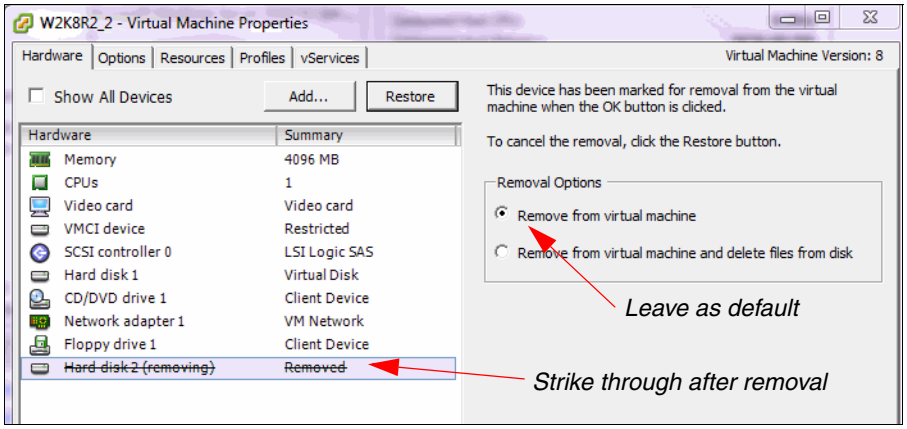


Figure 9-64 Raw Device Mapping after being removed

3. From the non XIV storage, map the migration volume to the XIV migration host, create the migration volume, and map that volume to the ESX/ESXi host. This is the same as a regular XIV data migration and is detailed in 9.4, “Data migration steps” on page 253. An important point is to take note of the LUN ID used to map the volume to the vSphere cluster and the Serial number of the volume. In Figure 9-65 it is `0x1a57`, and although not shown, it was mapped as LUN ID 4. If the serial number column is not visible, right-click in the headings area of the panel and add it.

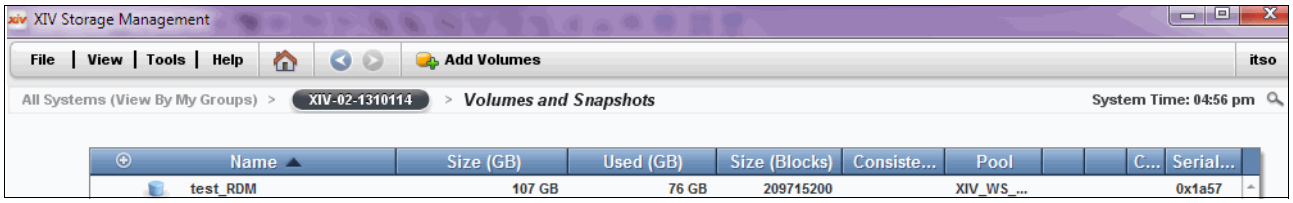


Figure 9-65 RDM after being migrated

4. From the vSphere Client, go to Hosts and Clusters and select your ESX/ESXi host. Select the **Configuration** tab and click **Storage**. Display **Devices** (rather than data stores) and select the **Recan All** option. If the device is correctly mapped from the XIV, a new volume is displayed. In step 3 you noted the LUN ID and serial number; in Figure 9-66 you can see a new LUN with an identifier that ends in `1a57` and that is LUN ID 4 (L4 in the Runtime Name).

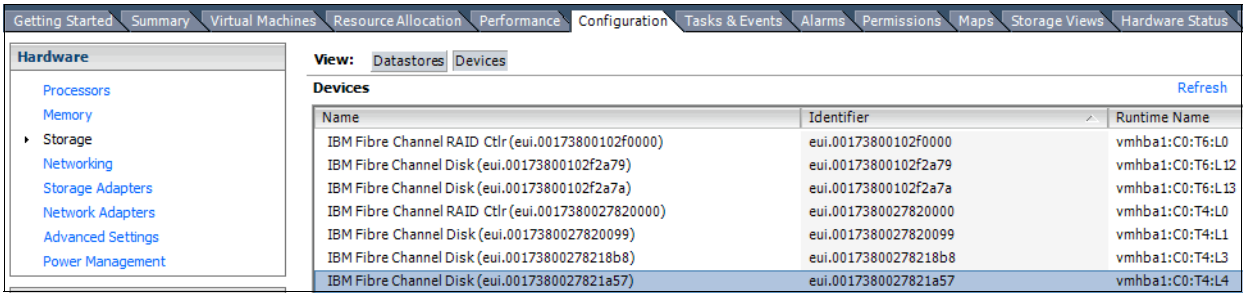


Figure 9-66 RDM detected by vSphere

5. Using the vSphere Client right-click the virtual machine and select **Edit Settings** to open the Virtual Machine Properties panel.
6. Select the option to **Add a Device** → **Hard Disk** and click **Next**.

7. Select the choice to **Use Raw Device Mappings** and click **Next**.
8. Select the disk with the correct Identifier, Path ID, and LUN ID and click **Next**.
9. Continue to make selections according to your standard policy (which might be the defaults) until you get the end of the options. Click **Finish**.
10. Power the virtual machine back on, or from within the guest operating system, scan for new devices to detect the disk. The process is now complete.

9.16 Sample migration

This section describes the sample migration we performed.

Using XIV DM to migrate an AIX file system from ESS 800 to XIV

In this example, we migrate a file system on an AIX host using ESS 800 disks to XIV. First, we select a volume group to migrate. In Example 9-16 we select a volume group called ESS_VG1. The **lsvg** command shows that this volume group has one file system mounted on /mnt/redbk. The **df -k** command shows that the file system is 20 GB in size and is 46% used.

Example 9-16 Selecting a file system

```
root@dolly:/mnt/redbk# lsvg -l ESS_VG1
ESS_VG1:
LV NAME             TYPE       LPs     PPs     PVs  LV STATE      MOUNT POINT
loglv00             jfs2log    1       1       1    open/syncd    N/A
fslv00              jfs2       20      20      3    open/syncd    /mnt/redbk
root@dolly:/mnt/redbk# df -k
Filesystem    1024-blocks      Free %Used    Iused %Iused Mounted on
/dev/fslv00      20971520  11352580   46%       17     1% /mnt/redbk
```

We now determine which physical disks must be migrated. In Example 9-17, we use the **lspv** commands to determine that hdisk3, hdisk4, and hdisk5 are the relevant disks for this VG. The **lsdev -Cc disk** command confirms that they are located on an IBM ESS 2105. We then use the **lscfg** command to determine the hardware serial numbers of the disks involved.

Example 9-17 Determine the migration disks

```
root@dolly:/mnt/redbk# lspv
hdisk1          0000d3af10b4a189                    rootvg          active
hdisk3          0000d3afbec33645                    ESS_VG1         active
hdisk4          0000d3afbec337b5                    ESS_VG1         active
hdisk5          0000d3afbec33922                    ESS_VG1         active
root@dolly:~/sddpcm# lsdev -Cc disk
hdisk0  Available 11-08-00-2,0 Other SCSI Disk Drive
hdisk1  Available 11-08-00-4,0 16 Bit LVD SCSI Disk Drive
hdisk2  Available 11-08-00-4,1 16 Bit LVD SCSI Disk Drive
hdisk3  Available 17-08-02     IBM MPIO FC 2105
hdisk4  Available 17-08-02     IBM MPIO FC 2105
hdisk5  Available 17-08-02     IBM MPIO FC 2105

root@dolly:/mnt# lscfg -vpl hdisk3 | egrep "Model|Serial"
        Machine Type and Model......2105800
        Serial Number...............00FFCA33
root@dolly:/mnt# lscfg -vpl hdisk4 | egrep "Model|Serial"
```

```
        Machine Type and Model......2105800
        Serial Number..............010FCA33
root@dolly:/mnt# lscfg -vpl hdisk5 | egrep "Model|Serial"
        Machine Type and Model......2105800
        Serial Number..............011FCA33
```

These volumes are currently allocated from an IBM ESS 800. In Figure 9-67 we use the ESS web GUI to confirm that the volume serial numbers match with those determined in Example 9-17 on page 322. The LUN IDs here are those used by ESS 800 with AIX hosts (IDs 500F, 5010, and 5011). They are not correct for the XIV and will be changed when we re-map them to the XIV.

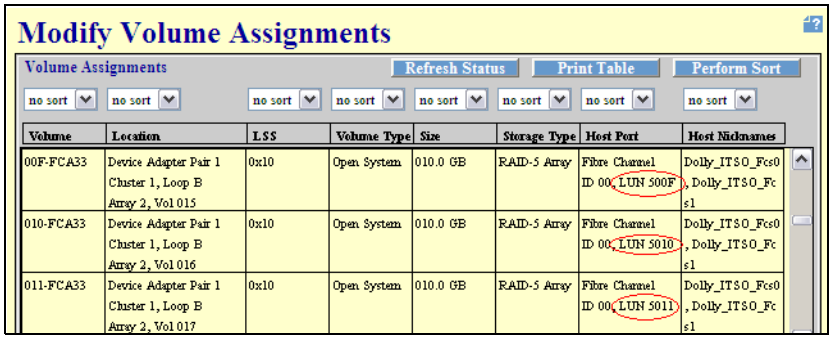


Figure 9-67 LUNs allocated to AIX from the ESS 800

Because we now know the source hardware we can create connections between the ESS 800 and the XIV and the XIV and Dolly (our host server). First, in Example 9-18 we identify the existing zones that connect Dolly to the ESS 800. We have two zones, one for each AIX HBA. Each zone contains the same two ESS 800 HBA ports.

Example 9-18 Existing zoning on the SAN Fabric

```
zone:  ESS800_dolly_fcs0
                10:00:00:00:c9:53:da:b3
                50:05:07:63:00:c9:0c:21
                50:05:07:63:00:cd:0c:21
 zone:  ESS800_dolly_fcs1
                10:00:00:00:c9:53:da:b2
                50:05:07:63:00:c9:0c:21
                50:05:07:63:00:cd:0c:21
```

We now create two new zones. The first zone connects the initiator ports on the XIV to the ESS 800. The second and third zones connect the target ports on the XIV to Dolly (for use after the migration). These are shown in Example 9-19. All six ports on the XIV clearly must have been cabled into the SAN fabric.

Example 9-19 New zoning on the SAN Fabric

```
zone:  ESS800_nextrazap
                50:05:07:63:00:c9:0c:21
                50:05:07:63:00:cd:0c:21
                50:01:73:80:00:23:01:53
                50:01:73:80:00:23:01:73
 zone:  nextrazap_dolly_fcs0
                10:00:00:00:c9:53:da:b3
                50:01:73:80:00:23:01:41
```

```
                50:01:73:80:00:23:01:51
 zone:  nextrazap_dolly_fcs1
                10:00:00:00:c9:53:da:b2
                50:01:73:80:00:23:01:61
                50:01:73:80:00:23:01:71
```

We create the migration connections between the XIV and the ESS 800. An example of using the XIV GUI to do this is in the bullet “Define target connectivity (Fibre Channel only):” on page 271.

In Example 9-20 we use the XCLI to define a target, then the ports on that target, then the connections between XIV and the target (ESS 800). Finally, we check that the links are *active=yes* and *up=yes*. We can use two ports on the ESS 800 because it is an active/active storage device.

Example 9-20 Connecting ESS 800 to XIV for migration using XCLI

```
>> target_define protocol=FC target=ESS800 xiv_features=no
Command executed successfully.
>> target_port_add fcaddress=50:05:07:63:00:c9:0c:21 target=ESS800
Command executed successfully.
>> target_port_add fcaddress=50:05:07:63:00:cd:0c:21 target=ESS800
Command executed successfully.
>> target_connectivity_define local_port=1:FC_Port:5:4
fcaddress=50:05:07:63:00:c9:0c:21 target=ESS800
Command executed successfully.
>> target_connectivity_define local_port=1:FC_Port:7:4
fcaddress=50:05:07:63:00:cd:0c:21 target=ESS800
Command executed successfully.
>> target_connectivity_list
Target Name   Remote Port        FC Port         IP Interface   Active   Up
ESS800        5005076300C90C21   1:FC_Port:5:4                  yes      yes
ESS800        5005076300CD0C21   1:FC_Port:7:4                  yes      yes
```

We define the XIV as a host to the ESS 800. In Figure 9-68 we have defined the two initiator ports on the XIV (with WWPNs that end in 53 and 73) as Linux (x86) hosts called Nextra_Zap_5_4 and NextraZap_7_4.

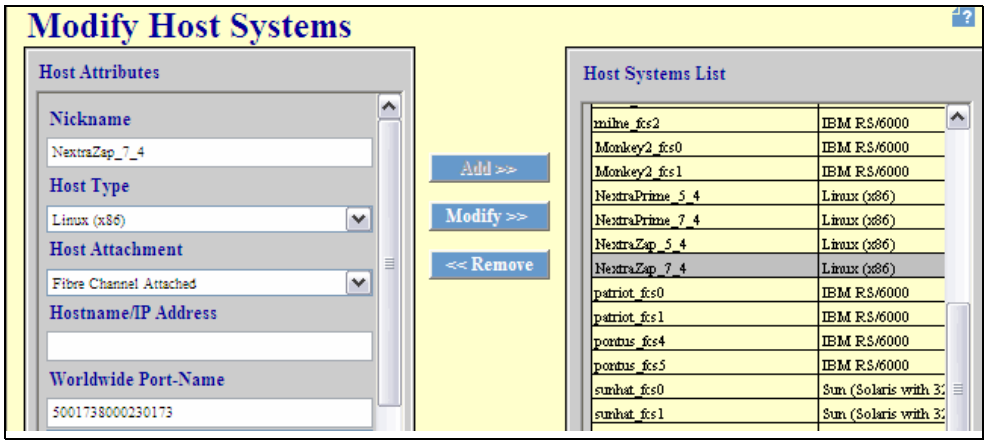


Figure 9-68 Define the XIV to the ESS 800 as a host

Finally, we can define the AIX host to the XIV as a host using the XIV GUI or XCLI. In Example 9-21 we use the XCLI to define the host and then add two HBA ports to that host.

Example 9-21 Define Dolly to the XIV using XCLI

```
>> host_define host=dolly
Command executed successfully.
>> host_add_port fcaddress=10:00:00:00:c9:53:da:b3 host=dolly
Command executed successfully.
>> host_add_port fcaddress=10:00:00:00:c9:53:da:b2 host=dolly
Command executed successfully.
```

After the zoning changes have been done and connectivity and correct definitions confirmed between XIV to ESS and XIV to AIX host, we take an outage on the volume group and related file systems that are going to be migrated. In Example 9-22 we unmount the file system, vary off the volume group, and then export the volume group. Finally, we use **rmdev** for the hdisk devices.

Example 9-22 Removing the non XIV file system

```
root@dolly:/# umount /mnt/redbk
root@dolly:/# varyoffvg ESS_VG1
root@dolly:/# exportvg ESS_VG1
root@dolly:/# rmdev -dl hdisk3
hdisk3 deleted
root@dolly:/# rmdev -dl hdisk4
hdisk4 deleted
root@dolly:/# rmdev -dl hdisk5
hdisk5 deleted
```

If the Dolly host no longer needs access to any LUNs on the ESS 800, we remove the SAN zoning that connects Dolly to the ESS 800. In Example 9-18 on page 323, those were the zones called ESS800_dolly_fcs0 and ESS800_dolly_fcs1.

We now allocate the ESS 800 LUNS to the XIV, as shown in Figure 9-69, where volume serials 00FFCA33, 010FCA33, and 011FCA33 have been unmapped from the host called Dolly and remapped to the XIV definitions called NextraZap_5_4 and NextraZap_7_4. We do not allow the volumes to be presented to both the host and the XIV. The LUN IDs in the Host Port column are correct for use with XIV because they start with zero and are the same for both NextraZap Initiator ports.

00F-FCA33	Device Adapter Pair 1 Cluster 1, Loop B Array 2, Vol 015	0x10	Open System	010.0 GB	RAID-5 Array	Fibre Channel ID 00, LUN 0000	NextraZap_5_4
010-FCA33	Device Adapter Pair 1 Cluster 1, Loop B Array 2, Vol 016	0x10	Open System	010.0 GB	RAID-5 Array	Fibre Channel ID 00, LUN 0001	NextraZap_5_4
011-FCA33	Device Adapter Pair 1 Cluster 1, Loop B Array 2, Vol 017	0x10	Open System	010.0 GB	RAID-5 Array	Fibre Channel ID 00, LUN 0002	NextraZap_5_4
00F-FCA33	Device Adapter Pair 1 Cluster 1, Loop B Array 2, Vol 015	0x10	Open System	010.0 GB	RAID-5 Array	Fibre Channel ID 00, LUN 0000	NextraZap_7_4
010-FCA33	Device Adapter Pair 1 Cluster 1, Loop B Array 2, Vol 016	0x10	Open System	010.0 GB	RAID-5 Array	Fibre Channel ID 00, LUN 0001	NextraZap_7_4
011-FCA33	Device Adapter Pair 1 Cluster 1, Loop B Array 2, Vol 017	0x10	Open System	010.0 GB	RAID-5 Array	Fibre Channel ID 00, LUN 0002	NextraZap_7_4

Figure 9-69 LUNs allocated to the XIV

We create the DMs and run a test on each LUN. The XIV GUI or XCLI could be used. In Example 9-23 the commands to create, test, and activate one of the three migrations are shown. We must run each command for hdisk3 and hdisk4 also.

Example 9-23 Creating one migration

```
> dm_define target="ESS800" vol=”dolly_hdisk3” lun=0 source_updating=yes create_vol=yes pool=AIX
Command executed successfully.
> dm_test vol=”dolly_hdisk3”
Command executed successfully.
> dm_activate vol=”dolly_hdisk3”
Command executed successfully.
```

After we create and activate all three migrations, the Migration panel in the XIV GUI looks as shown in Figure 9-70 (illustrations are based on a former version of the XIV GUI). The remote LUN IDs are 0, 1, and 2, which must match the LUN numbers seen in Figure 9-69 on page 325.

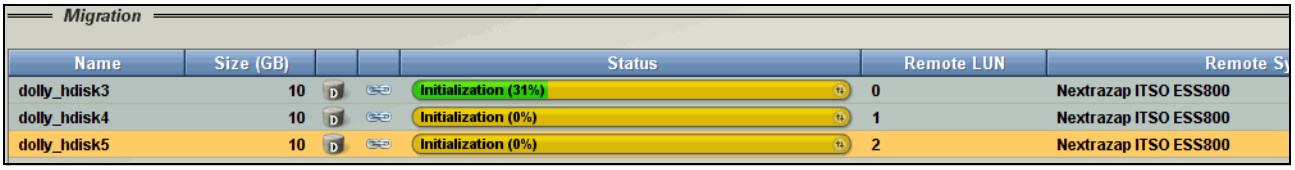


Figure 9-70 Migration has started

Now that the migration is started, we can map the volumes to the AIX host definition on the XIV, as shown in Figure 9-71, where the AIX host is called Dolly.

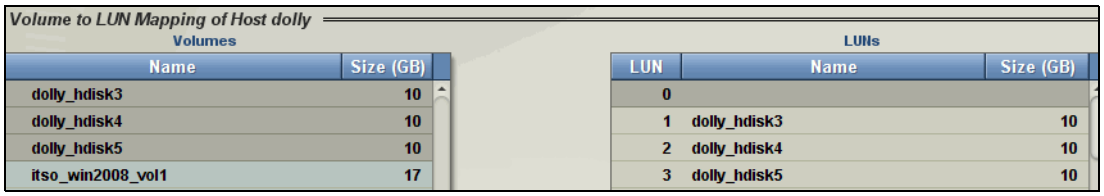


Figure 9-71 Map the XIV volumes to the host

Now we can bring the volume group back online. Because this AIX host was already using SDDPCM, we can install the XIVPCM (the AIX host attachment kit) at any time prior to the change. In Example 9-24, we confirm that SDDPCM is in use and that the XIV definition file set is installed. We then run **cfgmgr** to detect the new disks. We confirm that the disks are *visible* by using the **lsdev -Cc** disk command.

Example 9-24 Rediscovering the disks

```
root@dolly:~# lslpp -L | grep -i sdd
  devices.sddpcm.53.rte      2.2.0.4    C     F    IBM SDD PCM for AIX V53
root@dolly:/# lslpp -L | grep 2810
  disk.fcp.2810.rte          1.1.0.1    C     F    IBM 2810XIV ODM definitions
root@dolly:/# cfgmgr -l fcs0
root@dolly:/# cfgmgr -l fcs1
root@dolly:/# lsdev -Cc disk
hdisk1  Available 11-08-00-4,0 16 Bit LVD SCSI Disk Drive
hdisk2  Available 11-08-00-4,1 16 Bit LVD SCSI Disk Drive
hdisk3  Available 17-08-02     IBM 2810XIV Fibre Channel Disk
hdisk4  Available 17-08-02     IBM 2810XIV Fibre Channel Disk
hdisk5  Available 17-08-02     IBM 2810XIV Fibre Channel Disk
```

A final check before bringing the volume group back ensures that the Fibre Channel pathing from the host to the XIV is set up correctly. We can use the AIX **lspath** command against each hdisk, as shown in Example 9-25. In this example, the host can connect to port 2 on each of the XIV modules 4, 5, 6, and 7 (which is confirmed by checking the last two digits of the WWPN).

Example 9-25 Using the lspath command

```
root@dolly:~/# lspath -l hdisk5 -s available -F"connection:parent:path_status:status"
5001738000230161,3000000000000:fscsi1:Available:Enabled
5001738000230171,3000000000000:fscsi1:Available:Enabled
5001738000230141,3000000000000:fscsi0:Available:Enabled
5001738000230151,3000000000000:fscsi0:Available:Enabled
```

We can also use a script provided by the XIV Host Attachment Kit for AIX, called **xiv_devlist**. An example of the output is shown in Example 9-26.

Example 9-26 Using xiv_devlist

```
root@dolly:~# xiv_devlist
XIV devices
===========
Device               Vol Name     XIV Host     Size    Paths XIV ID   Vol ID
-------------------------------------------------------------------------------
hdisk3               dolly_hdisk3 dolly        10.0GB  4/4   MN00023  8940
hdisk4               dolly_hdisk4 dolly        10.0GB  4/4   MN00023  8941
hdisk5               dolly_hdisk5 dolly        10.0GB  4/4   MN00023  8942

Non XIV devices
===============
Device               Size    Paths
-----------------------------------
hdisk1               N/A     1/1
hdisk2               N/A     1/1
```

We can also use the XIV GUI to confirm connectivity by going to the **Hosts and Clusters** → **Host Connectivity** panel. An example is shown in Figure 9-72, where the connections match those seen in Example 9-25.

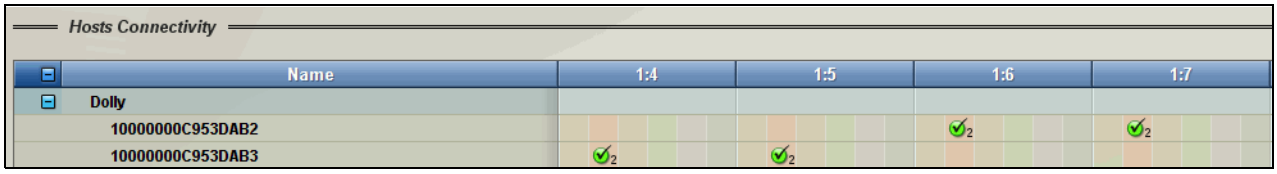


Figure 9-72 Host Connectivity panel

Having confirmed that the disks have been detected and that the paths are good, we can now bring the volume group back online. In Example 9-27 we import the VG, and confirm that the PVIDs match those in Example 9-17 on page 322, and then mount the file system.

Example 9-27 Bring the VG back online

```
root@dolly:/# /usr/sbin/importvg -y'ESS_VG1' hdisk3
ESS_VG1
root@dolly:/# lsvg -l ESS_VG1
ESS_VG1:
LV NAME             TYPE       LPs     PPs     PVs  LV STATE      MOUNT POINT
loglv00             jfs2log    1       1       1    closed/syncd  N/A
fslv00              jfs2       20      20      3    closed/syncd  /mnt/redbk
root@dolly:/# lspv
hdisk1          0000d3af10b4a189                    rootvg          active
hdisk3          0000d3afbec33645                    ESS_VG1         active
hdisk4          0000d3afbec337b5                    ESS_VG1         active
hdisk5          0000d3afbec33922                    ESS_VG1         active
root@dolly:/# mount /mnt/redbk
root@dolly:/mnt/redbk# df -k
Filesystem    1024-blocks      Free %Used    Iused %Iused Mounted on
/dev/fslv00      20971520  11352580   46%       17     1% /mnt/redbk
```

After the sync is complete it is time to delete the migrations. Do not leave the migrations in place any longer than they need to be. We can use multiple selection to perform the deletion, as shown in Figure 9-73, taking care to delete and not deactivate the migration.

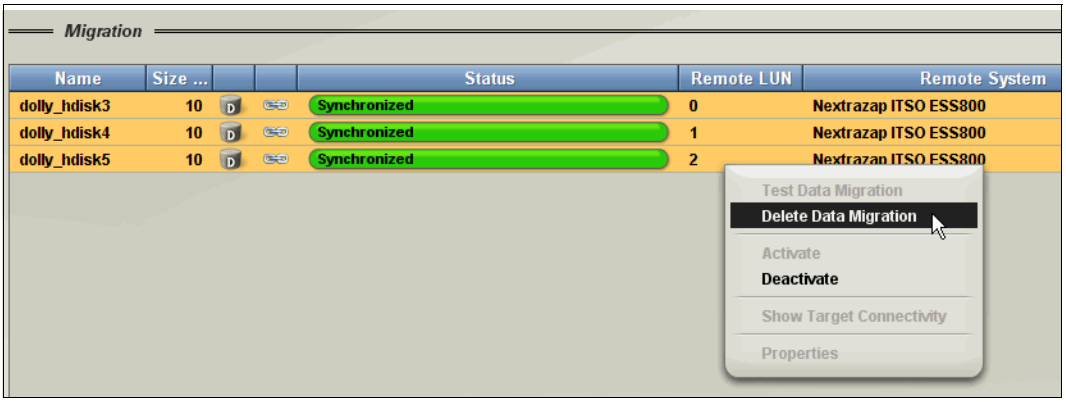


Figure 9-73 Deletion of the synchronized data migration

Now at the ESS 800 web GUI we can unmap the three ESS 800 LUNs from the Nextra_Zap host definitions. This frees up the LUN IDs to be reused for the next volume group migration.

After the migrations are deleted, a final suggested task is to re-size the volumes on the XIV to the next 17 GB cutoff. In this example we migrate ESS LUNs that are 10 GB in size. However, the XIV commits 17 GB of disk space because all space is allocated in 17 GB portions. For this reason it is better to resize the volume on the XIV GUI from 10 GB to 17 GB so that all the allocated space on the XIV is available to the operating system. This presumes that the operating system can tolerate a LUN size growing, which in the case of AIX is true.

We must unmount any file systems and vary off the volume group before we start. Then we go to the volumes section of the XIV GUI, right-click to select the 10 GB volume, and select the **Resize** option. The current size is displayed.

In Figure 9-74 the size is shown in 512-byte blocks because the volume was automatically created by the XIV, based on the size of the source LUN on the ESS 800. If we multiply 19531264 by 512 bytes we get 10,000,007,168 bytes, which is 10 GB.

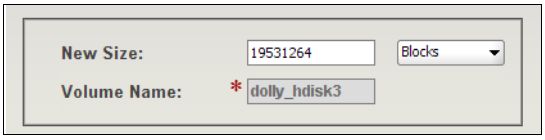


Figure 9-74 Starting volume size in blocks

We change the sizing methodology to GB and the size immediately changes to 17 GB, as shown in Figure 9-75. If the volume was already larger than 17 GB, it changes to the next interval of 17 GB. For example, a 20 GB volume shows as 34 GB.

Figure 9-75 Size changed to GB

A warning message indicates that the volume is increasing in size. Click **OK** to continue.

Now the volume is really 17 GB and no space is wasted on the XIV. The new size is shown in Figure 9-76.

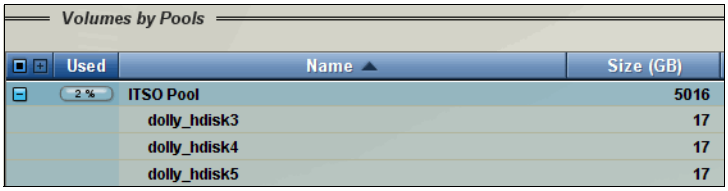


Figure 9-76 Resized volumes

We vary on the VG again to update AIX that the volume size has changed. Example 9-28 shows importing the VG, which detects that the source disks have grown in size. Then, we run the **chvg -g** command to grow the volume group, and confirm that the file system can still be used.

Example 9-28 Importing larger disks

```
root@dolly:~# /usr/sbin/importvg -y'ESS_VG1' hdisk3
0516-1434 varyonvg: Following physical volumes appear to be grown in size.
        Run chvg command to activate the new space.
        hdisk3        hdisk4        hdisk5
ESS_VG1
root@dolly:~# chvg -g ESS_VG1
root@dolly:~# mount /mnt/redbk
root@dolly:/mnt/redbk# df -k
Filesystem    1024-blocks      Free %Used    Iused %Iused Mounted on
/dev/fslv00      20971520  11352580   46%       17     1% /mnt/redbk
```

We can now resize the file system to take advantage of the extra space. In Example 9-29 the original size of the file system in 512-byte blocks is shown.

Example 9-29 Displaying the current size of the file system

```
Change/Show Characteristics of an Enhanced Journaled File System

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  File system name                                    /mnt/redbk
  NEW mount point                                    [/mnt/redbk]
  SIZE of file system
         Unit Size                                    512bytes
         Number of units                             [41943040]
```

We change the number of 512-byte units to 83886080 because this is 40 GB in size, as shown in Example 9-30.

Example 9-30 Growing the file system

```
SIZE of file system
     Unit Size                512bytes
+
          Number of units                            [83886080]
```

The file system has grown. Example 9-31 shows that the file system has grown from 20 GB to 40 GB.

Example 9-31 Displaying the enlarged file system

```
root@dolly:~# df -k
/dev/fslv00      41943040  40605108    4%        7     1% /mnt/redbk
```

10

Using Tivoli Storage Productivity Center for Replication

The IBM Tivoli Storage Productivity Center for Replication[1] is an automated solution providing a management front end to Copy Services over many IBM products.

Tivoli Productivity Center for Replication can help manage Copy Services (Snapshot and Mirrors) for the XIV storage system, and this is the focus for this chapter. At the time of writing, the latest Tivoli Productivity Center for Replication version is 5.1.

See the following sources for more information:

- Information about implementing and managing Tivoli Productivity Center for Replication configurations:

  http://www-947.ibm.com/support/entry/portal/documentation/software/tivoli/tivoli_storage_productivity_center

- Supported Storage Products Matrix website:

  http://www.ibm.com/support/docview.wss?uid=swg27027303

- *IBM Tivoli Storage Productivity Center V4.2 Release Guide,* SG24-7894

Note: Tivoli Storage Productivity Center for Replication version 4.2.2.1 or later is required for XIV Gen3 systems.

[1] Also referred to as “Tivoli Productivity Center for Replication” in this chapter.

© Copyright IBM Corp. 2014. All rights reserved.

10.1 IBM Tivoli Productivity Center family

The IBM Tivoli Storage Productivity Center is a wide suite of software products. It is designed to support customers in monitoring and managing their storage environments.

The design and development emphasis for Tivoli Productivity Center is on scalability and standards. The approach based on open standards allows Tivoli Productivity Center to manage any equipment or solution implementations that follow the same open standards.

Tivoli Productivity Center products provide a single source and single management tool to cover the tasks of the storage manager or network manager in their daily business.

- Tivoli Productivity Center for Disk is software based on open standard interfaces to query, gather, and collect all available data necessary for performance management.
- Tivoli Productivity Center for Data focuses on data management and addresses aspects related to information life-cycle management.
- Tivoli Productivity Center for Fabric is a management tool to monitor and manage a SAN fabric.

Tivoli Productivity Center for Replication is offered in several editions:

- Tivoli Productivity Center for Replication, Basic Edition (5608-TRA), which includes support for XIV Snapshots
- Tivoli Productivity Center for Replication, Standard Edition for Two Sites Business Continuity (BC), (5608-TRB), which includes support for the following items:
  - XIV Snapshots
  - Planned and unplanned failover and failback for XIV Asynchronous and Synchronous Mirroring, known inside Tivoli Productivity Center for Replication as Global Mirror and Metro Mirror respectively
  - High availability (with two Tivoli Productivity Center for Replication servers)

10.2 What Tivoli Productivity Center for Replication provides

Tivoli Storage Productivity Center for Replication is designed to help administrators manage XIV Copy Services. This also applies not only to the XIV but also to the DS8000, SAN Volume Controller, Storwize Family, and Storwize V7000U.

Tivoli Productivity Center for Replication simplifies management of Copy Services:

- By automating administration and configuration of Copy Services functions with wizard-based sessions and copy set definitions.
- By providing simple operational control of Copy Services tasks, which includes starting, suspending, and resuming Copy Services Tasks.
- The optional high availability feature allows you to continue replication management even when one Tivoli Productivity Center for Replication server goes down.

Tivoli Productivity Center for Replication provides management of the following XIV functions:

- Snapshots
- Synchronous mirrors, known as inside Tivoli Productivity Center for Replication as Metro Mirror
- Asynchronous mirrors, known as Global Mirror

Important: At the time of writing, the following limitations exist:

- All XIV storage pools and volumes must be configured within the XIV GUI or XCLI.
- All mirroring connectivity must be configured within the XIV GUI or XCLI.

Figure 10-1 is from the Tivoli Productivity Center for Replication GUI showing the variety of session types, or Copy Services functions, supported on the XIV.

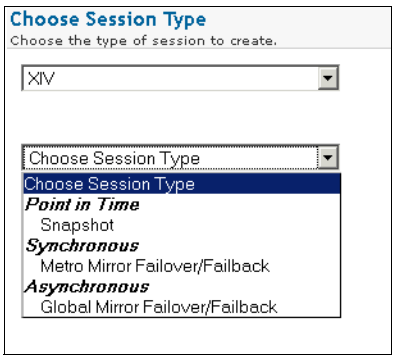


Figure 10-1 XIV Copy Services Functions supported by Tivoli Productivity Center for Replication

10.3 Supported operating system platforms

Currently, the Tivoli Productivity Center for Replication server can run on the following commercial operating systems:

- Windows 2008 Server Editions, including Windows 2008 R2 Editions
- Red Hat Enterprise Linux 5 including Advanced Platform
- VMware ESX and ESXi 3.0.x, 3.5.x, 4.0.x, and 4.1.x with VMs running the Windows and Linux versions listed in the previous two bullets
- AIX 6.1, TL 4, SP5 or later, AIX 7.1
- IBM z/OS® V1.10, V1.11, V1.12

For a Tivoli Productivity Center for Replication Two Site BC configuration that involves two Tivoli Productivity Center for Replication servers, it is possible to run Tivoli Productivity Center for Replication under two separate operating systems.

See the most recent requirements:

https://www.ibm.com/support/docview.wss?rs=40&uid=swg21386446&context=SSBSEX&cs=utf-8&lang=en&loc=en_US

10.4 Copy Services terminology

Although XIV Copy Services terminology was discussed previously in this book, we have included this section to review the definitions in the context of Tivoli Productivity Center for Replication. This section also includes a brief review of XIV concepts as they relate to Tivoli Productivity Center for Replication.

Session

A *session* is a metadata descriptor that defines the type of Copy Service operation to be performed on the specified volumes contained within itself. Each session is of a single, particular type: Snapshot, Metro Mirror, or Global Mirror. These session types are discussed and illustrated in more detail later in this chapter.

Snapshot

A *snapshot* is a point-in-time copy of a volume's data. Snapshots make use of pointers and do not necessarily copy all the data to the second instance of a volume.

Remote mirror

The remote mirroring function of the XIV Storage System provides a real-time copy between two or more XIV storage systems supported over Fibre Channel (FC) or iSCSI links. This feature provides a method to protect data from individual site failures.

Remote mirroring can be a synchronous copy solution where write operations are completed on both copies (local and remote sites) before they are considered to be complete. Tivoli Productivity Center for Replication considers XIV synchronous mirrors as *Metro Mirrors*.

Remote mirroring can also be an asynchronous solution in which consistent sets of data are copied to the remote location at specified intervals and host I/O operations are complete after writing to the primary. This is typically used for long distances between sites. Tivoli Productivity Center for Replication considers XIV's asynchronous mirrors as *Global Mirrors*.

Consistency groups and Tivoli Productivity Center for Replication use of sessions

Tivoli Productivity Center for Replication uses XIV consistency groups (CG) for all three of the session types mentioned. These CGs are both created and named by Tivoli Productivity Center for Replication, using the session name that is entered using the session wizard at creation time.

Important: When Tivoli Productivity Center for Replication attempts to create a consistency group, using the supplied session name, it might already exist on the XIV. Tivoli Productivity Center for Replication will then attempt to use the supplied name with an "_001" suffix. It will keep trying in a similar fashion "_00x" until x=30, at which point Tivoli Productivity Center for Replication will fail the operation.

If this naming operation fails, than Tivoli Productivity Center for Replication will fail to create the consistency group and return an error on the session wizard.

Volumes

With Tivoli Productivity Center for Replication, the role of a volume within XIV Copy Services has been renamed to be more generic. It is the most granular of all the Tivoli Productivity Center for Replication elements. Volumes belong to a *copy set*.

Terminology is slightly different from what you might be used to. For example, XIV uses a primary or source volume at the primary site and a secondary or target volume at the slave or target end. Such a pair of volumes is now called a copy set.

Host volume

A host volume is identical to what is called a primary or source volume in Copy Services. The host designation represents the volume functional role from an application point of view. It is usually connected to a host or server and receives read, write, and update application I/Os.

When a host volume becomes the target volume of a Copy Services function, it is usually no longer accessible from the application host. FlashCopy target volumes can be considered as an exception.

Target volume

A target volume is what was also usually designated as a secondary volume. It receives data from a host volume or another intermediate volume.

Volume protection

Tivoli Productivity Center for Replication considers this concept as a method to mark storage volumes as protected if you do not want those volumes used for replication (for example, when a volume on the target site contains data that is not currently in a relationship, but you want to ensure that data is not overwritten).

Copy set

A *copy set* contains the volumes that are part of a Copy Services relationship. Tivoli Productivity Center for Replication uses copy sets which become *consistency groups* inside the XIV interface. A session defines the type of operation to be performed on the volumes contained within the session. Sessions contain copy sets.

10.4.1 Copy sets

A copy set in Tivoli Productivity Center for Replication is a set of volumes that have the same XIV copy services operation, such as synchronous mirroring. In XIV terms this is, for example, a master volume and its corresponding target, contained within a Tivoli Productivity Center for Replication copy set.

Figure 10-2 shows three pairs. In Tivoli Productivity Center for Replication each pair is considered as a copy set.

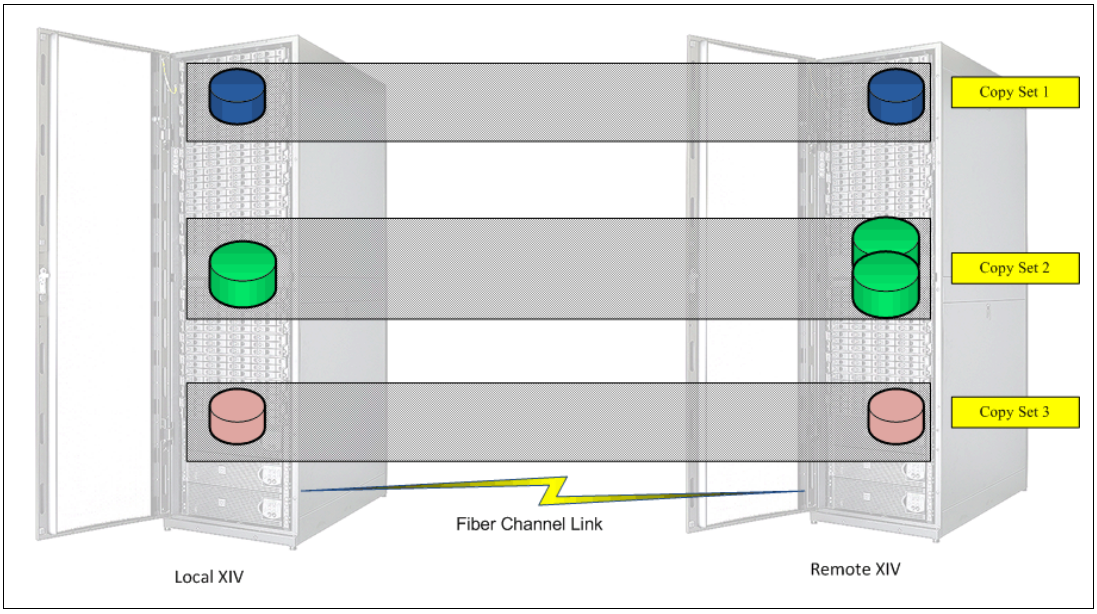


Figure 10-2 Tivoli Productivity Center for Replication: Metro Mirror copy sets, known as XIV synchronous mirrors

10.4.2 Sessions

Tivoli Productivity Center for Replication uses *sessions* for its primary operations. A session is a logical concept that gathers multiple copy sets, representing a group of volumes, with the requirement to provide consistent data within the scope of all involved volumes. Commands and processes performing against a session apply these actions to all copy sets within the session.

Figure 10-3 shows an example of multiple sessions between two sites. Each session can have more than one pair of volumes. The example further assumes that a Metro Mirror relation is established to replicate data between both sites.

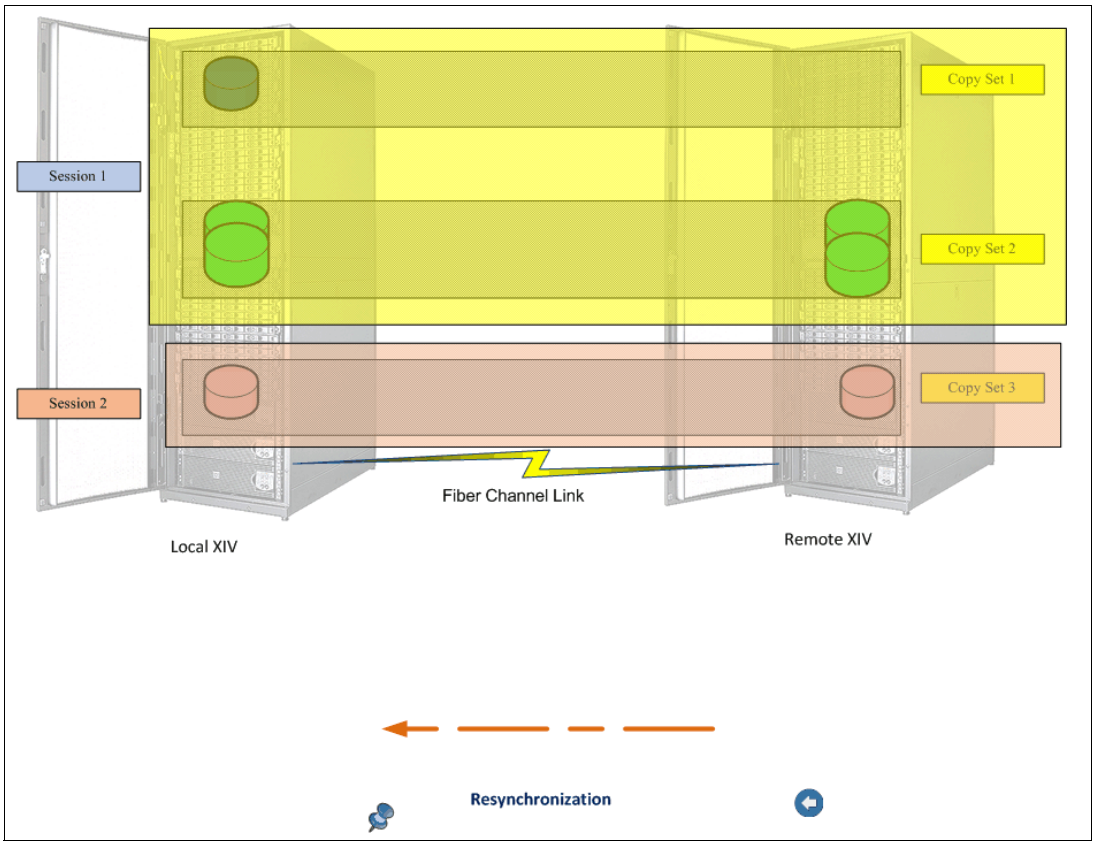


Figure 10-3 Tivoli Productivity Center for Replication: Session concept

Figure 10-3 shows Metro Mirror primary volume H1 from copy set 1, and volumes H2 (copy set 2), with their corresponding Metro Mirror secondary volumes grouped together in a session, *Session 1*.

Metro Mirror primary volumes H3, along with their counterparts H3 in the Copy Set 3, belong to a different session, *Session 2*.

Note: All application-dependent copy sets must belong to the same session to guarantee successful management and to provide consistent data across all involved volumes within a session. Volumes or copy sets that require consistent data should be grouped in one consistency group, which can be put into one session.

10.5 Session states

A session contains a group of copy sets (XIV volume or volume pairs) that belong to a certain application. You can also consider it as a collection of volumes that belong to a certain application or system, with the requirement for consistency.

Such a session can be in one of the following states:

- Defined: A session is defined and might already contain copy sets or still have no copy sets assigned yet. However, a defined session is not yet started.
- Flashing: Data copying is temporarily suspended while a consistent practice copy of data is being prepared on site 2.
- Preparing: The session started already and is in the process of initializing, which might be, for example, in the case of a first full initial copy for a Metro Mirror. It could also just re-initialize, which, for example, might be the case for a resynchronization of a previously suspended Metro Mirror. After the initialization is complete, the session state changes to prepared.
- Prepared: All volumes within the session completed the initialization process.
- Suspending: This is a transition state caused by either a suspend command or any other suspend trigger, such as an error in the storage subsystem or loss of connectivity between sites. Eventually the process to suspend copy sets ends and copying has stopped, which is indicated by a session state of suspended.
- Suspended: Replicating data from site 1 to site 2 has stopped.
- Recovering: A session is about to recover.
- TargetAvailable: The recover command has completed and the target volumes are write enabled and available for application I/Os. An additional recoverable flag indicates whether the data is consistent and recoverable.

Important: Do not manage through other software, Copy Service volume pairs that are already managed through Tivoli Productivity Center for Replication.

10.6 System and connectivity overview

Tivoli Productivity Center for Replication is an external application and its software runs on a dedicated server, or two, for a high availability configuration. Figure 10-4 presents a graphical overview.

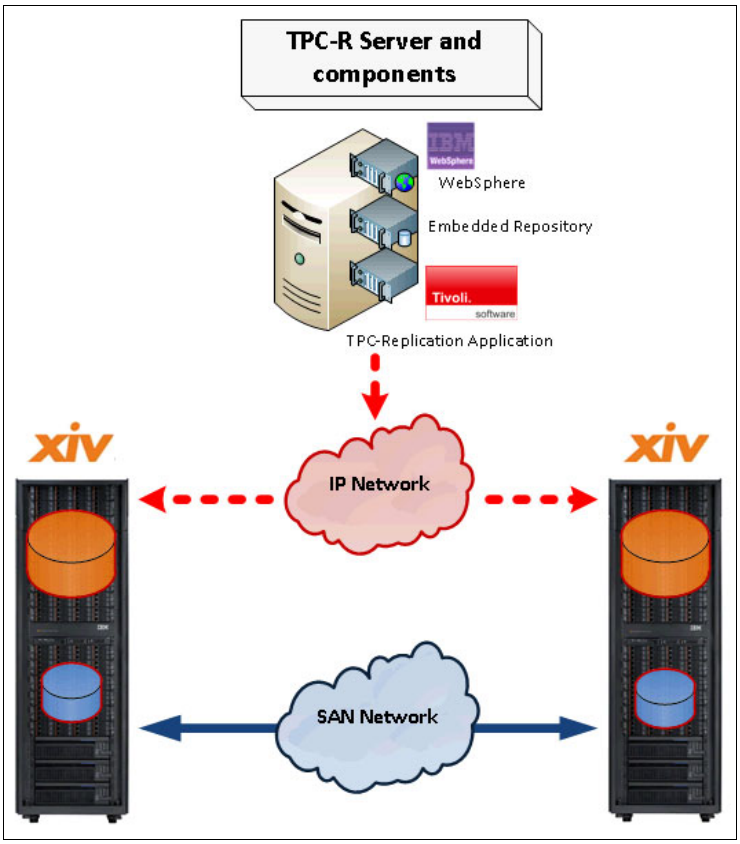


Figure 10-4 Tivoli Productivity Center for Replication system overview

Besides the Tivoli application code, Tivoli Productivity Center for Replication Server contains the IBM WebSphere® software, and an embedded repository.

IP communication services utilize native API commands and also an event listening capability to react to events from the XIV. This provides pre-established scripts that are going to be triggered by a corresponding trap message. This also includes a capability to distinguish among the various storage servers that can be managed by the Tivoli Productivity Center for Replication server, such as SAN Volume Controller and the DS8000. This approach has the potential to be enhanced as storage servers change over time without touching the actual functional code.

The actual operational connectivity between the Tivoli Productivity Center for Replication server and the XIV is based on Ethernet networks. Tivoli Productivity Center for Replication data transfers utilize the SAN network and connectivity established between XIV's for all data transfers.

10.7 Monitoring

Tivoli Productivity Center for Replication always uses the consistency group attribute when you define paths between a primary and a secondary storage server. This provides Tivoli Productivity Center for Replication with the capability to deactivate a Metro Mirror configuration when an incident happens, and guarantee consistent data at the secondary or backup site.

Tivoli Productivity Center for Replication server listens to incidents from the XIV and takes action when notified of a replication error from that specific storage system.

Figure 10-5 illustrates a replication error in a session. The Tivoli Productivity Center server receives a corresponding message from the XIV. The Tivoli Productivity Center for Replication server then issues a deactivate command to all volumes that are part of the concerned session. This implies a suspension of all volume pairs or copy sets that belong to this session. During this process, write I/Os are held until the process ends and the Tivoli Productivity Center server communicates to the XIV to continue processing write I/O requests to the concerned primary volumes in the session.

After that, write I/O can continue to the suspended primary volumes. However, both sites are not in sync any longer, but the data on the secondary site is consistent (power drop consistent).

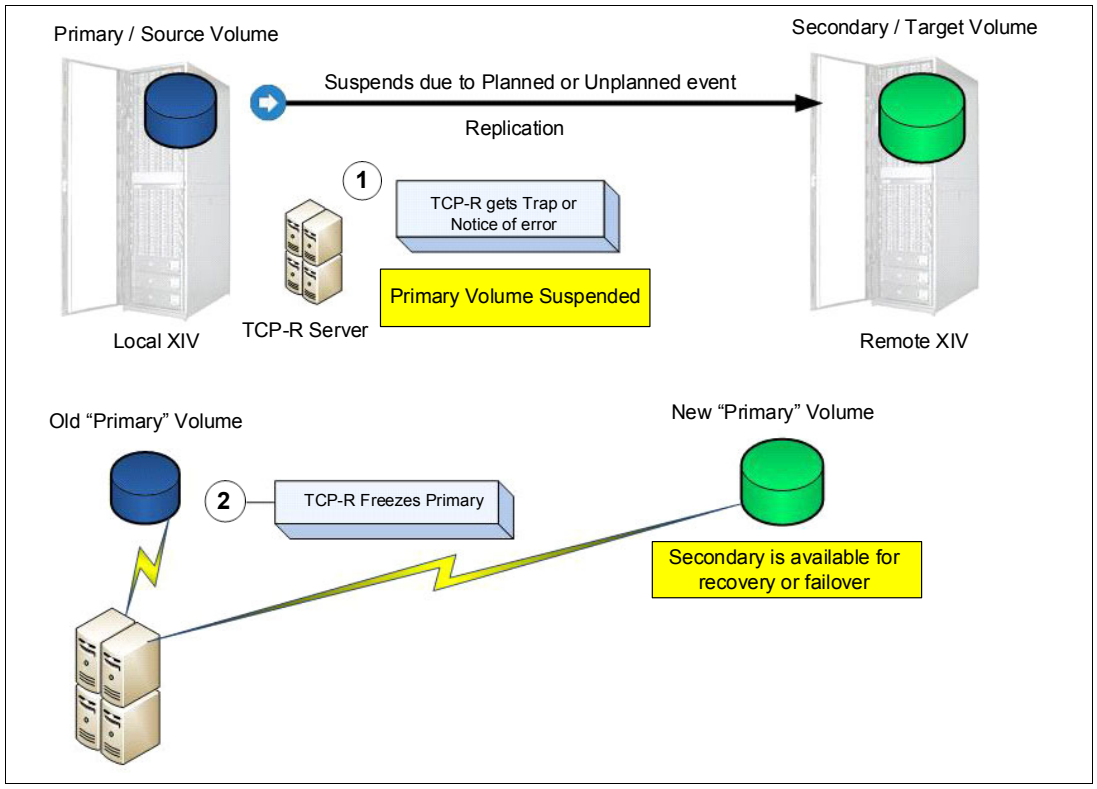


Figure 10-5 Tivoli Productivity Center for Replication server taking action during outage and freeze

10.8 Web interface

Tivoli Productivity Center for Replication provides a graphical user interface to manage and monitor any Copy Services configuration and Copy Services operations. This GUI is browser-based and does not rely on any other product.

The panel structure allows you to quickly transition to the various sections through a hyperlink-based menu.

Important: The Tivoli Productivity Center for Replication GUI makes use of pop-up browser windows for many of the wizard-based steps. Ensure your browser is set to enable these pop-ups to display.

10.8.1 Connecting to Tivoli Productivity Center for Replication GUI

You connect to the GUI by specifying the IP address of the Tivoli Storage Productivity Center for Replication server in the web browser. This opens the login panel, as shown in Figure 10-6. When you sign out from the server, the same panel is also displayed.

Tivoli.
IBM.
Tivoli Storage Productivity Center for Replication
User Name:
cliadmin
Password:
Version: 4.2.2
Build: j20110829-1206
Login
Connected to: Stops
Switch to alternate server: x3550-lab-6.mainz.de.ibm.com
Unsupported browser version detected. Some portions of the user interface may not render properly.
License Material - Property of IBM Corp. © IBM Corporation and other(s) 2009. IBM is a registered trademark of the IBM Corporation in the United States, other countries, or both.

Figure 10-6 Launch the Tivoli Storage Productivity Center for Replication server GUI

Specify a user ID as a text string in the UserID field and a password in the Password hidden text field. User IDs and passwords must have been previously defined and set up in the Tivoli Productivity Center for Replication server system.

10.8.2 Health Overview panel

After logging in to the Tivoli Storage Productivity Center for Replication server, the Health Overview panel opens, as shown in Figure 10-7 on page 341.

The Health Overview panel gives an overall summary of Tivoli Productivity Center for Replication system status. It shows information similar to what is also displayed in the small panel in the lower left corner of the window. This small health overview box in the lower left corner is always present.

However, the Health Overview panel provides more details. First, it provides overall status of the following items:

- Sessions
- Connected storage subsystems
- Host systems
- Management servers

Figure 10-7 Health Overview panel

Figure 10-7 shows that all sessions are in normal status and working fine. There is no high availability server environment, and one or more storage servers cannot be reached by the Tivoli Productivity Center for Replication server, given that they were defined before the Tivoli Productivity Center for Replication server was installed.

The upper left of the panel provides a list of Tivoli Productivity Center for Replication sections that you can use to manage various aspects of a Copy Services environment:

- Health Overview

  This is the currently displayed panel, as Figure 10-7 shows.

- Sessions

  This hyperlink brings you to the application that manages all sessions. This is the application that you will use the most.

- Storage Subsystems

  This is where you start when you define storage servers to the Tivoli Productivity Center for Replication server that are going to be used for Copy Services.

- Management Servers

  This link leads you to the application that manages the Tivoli Productivity Center for Replication server configuration.

- Advanced Tools

  Here you can trigger to collect diagnostic information or set the refresh cycle for the displayed data.

- Console

  This link opens a log that contains all activities that the user performed and their results.

10.8.3 Sessions panel

The Sessions panel (Figure 10-8) shows all sessions within the Tivoli Productivity Center for Replication server.

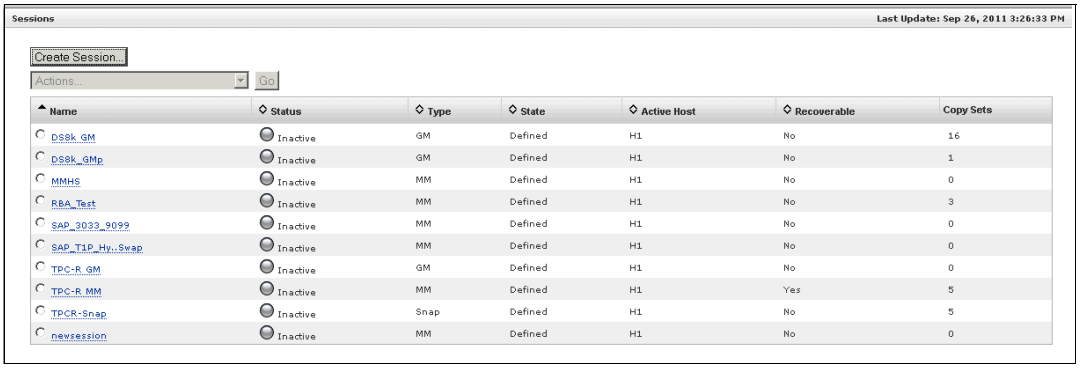

Sessions
Last Update: Sep 26, 2011 3:26:33 PM
Create Session...
Actions...
Go
Name
Status
Type
State
Active Host
Recoverable
Copy Sets
DS8k_GM Inactive GM Defined H1 No 16
DS8k_GMp Inactive GM Defined H1 No 1
MMHS Inactive MM Defined H1 No 0
RBA_Test Inactive MM Defined H1 No 3
SAP_3033_9099 Inactive MM Defined H1 No 0
SAP_T1P_Hy..Swap Inactive MM Defined H1 No 0
TPC-R_GM Inactive GM Defined H1 No 0
TPC-R_MM Inactive MM Defined H1 Yes 5
TPCR-Snap Inactive Snap Defined H1 No 5
newsession Inactive MM Defined H1 No 0

Figure 10-8 Sessions overview

Each session consists of several copy sets that can be distributed across XIV storage systems. The session name functions as a token that is used to apply any action against the session or all the volumes that belong to that session.

You first select a session and then chose the action you want to perform against that session.

10.8.4 Storage Subsystems panel

The Storage Subsystems panel (Figure 10-9) displays all storage subsystems defined in Tivoli Productivity Center for Replication.

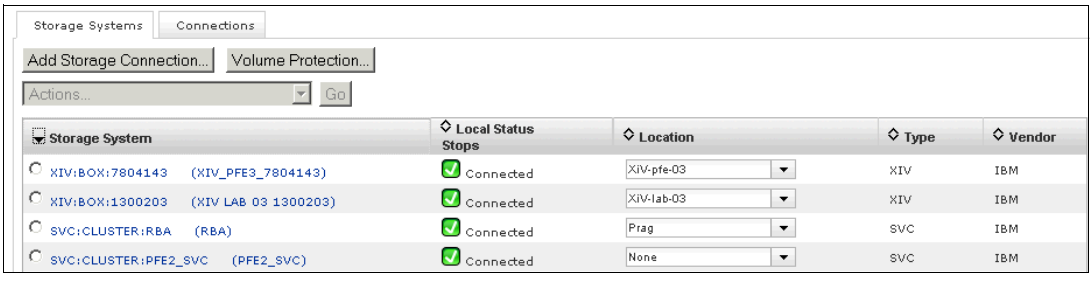


Figure 10-9 Storage GUI panel

10.9 Defining and adding XIV storage

This section describes how to create and modify a storage connection. Starting from the Tivoli Productivity Center for Replication Storage Subsystems panel, follow these steps:

1. Click **Add Storage Connection**.
2. The Add Storage System wizard opens (Figure 10-10). Select the **XIV** radio button, and click **Next.**

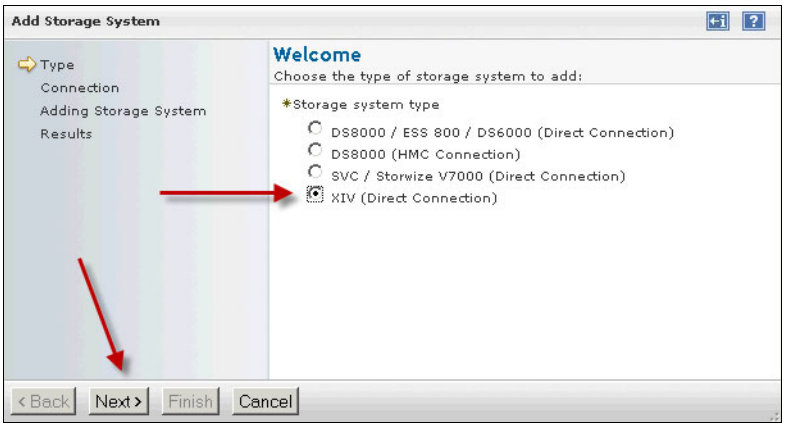


Figure 10-10 Adding the XIV to Tivoli Productivity Center for Replication

3. On the Connection window (Figure 10-11), enter the appropriate information:
   - XIV IP Address or Domain Name (Tivoli Productivity Center for Replication auto-populates the other IPs assigned.)
   - Username and Password for an XIV Admin level account

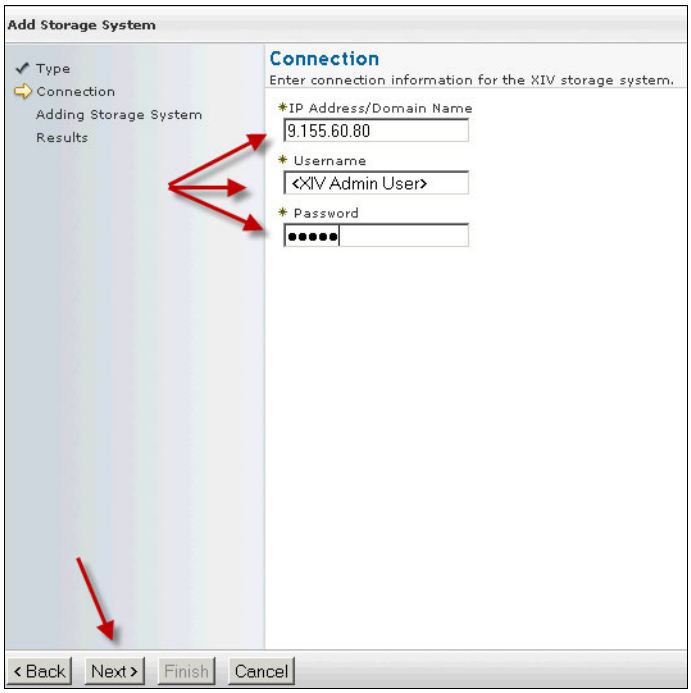


Figure 10-11 Specify one of the XIV's credentials

4. Click **Next**. The supplied credentials will be used to add the XIV storage system to the Tivoli Productivity Center for Replication repository.
5. The results are displayed in the next window (Figure 10-12). Click **Finish** to complete using the wizard.

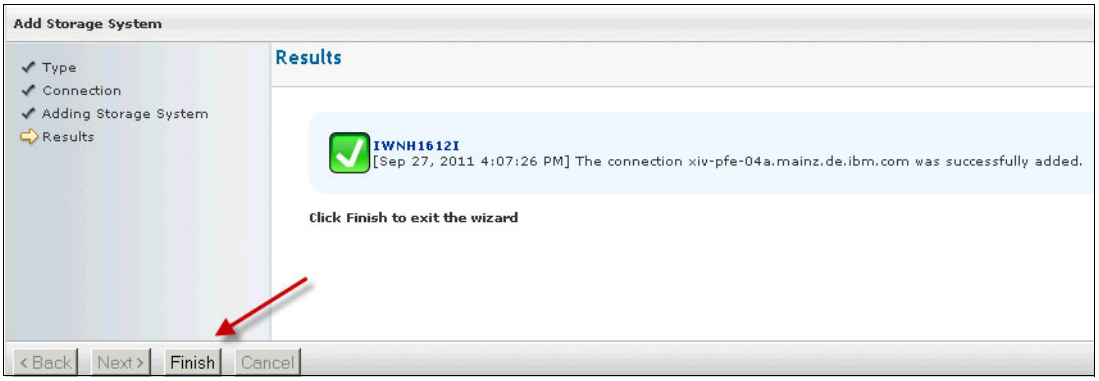
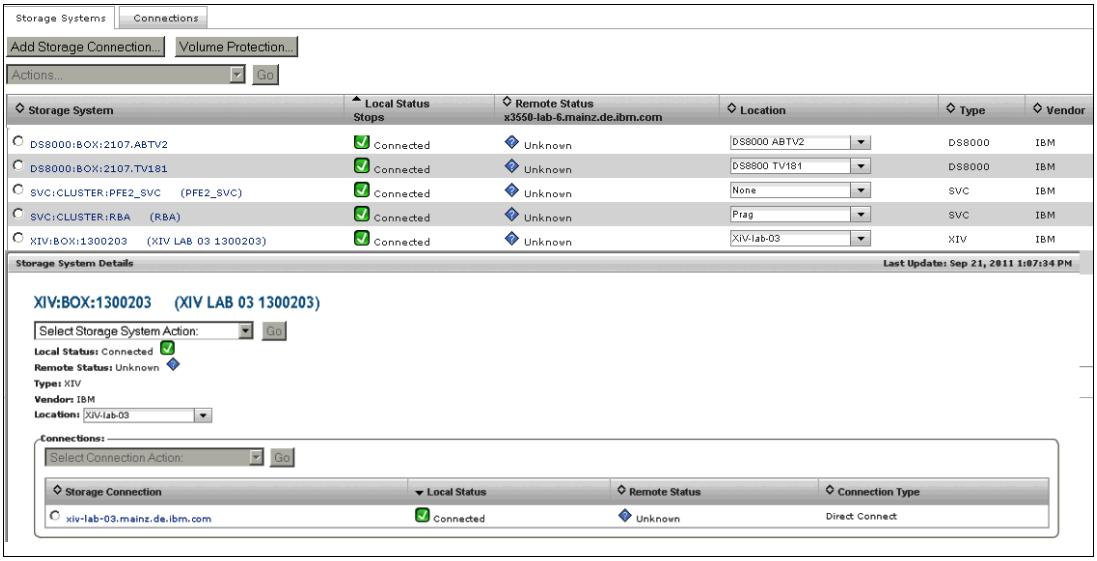


Figure 10-12 Step 3 Tivoli Productivity Center for Replication wizard with XIV added successfully

To modify an existing storage connection, click its name; the panel shown in Figure 10-13 on page 345 opens. You can change the site definition and add, modify, or remove the connection by making the appropriate selections and entries.



10.10 XIV snapshots

To use Tivoli Productivity Center for Replication for any type of XIV Copy Services, including snapshots, you first need to create and define a new session, add the XIV volumes (referred to as copy sets) in the new session, and activate the session.

As noted previously, both the XIV pool and volumes must be defined using the XIV GUI or CLI before using Tivoli Productivity Center for Replication.

We explore this entire process and flow, in greater detail next.

10.10.1 Defining a session for XIV snapshots

Use the following steps to define a Tivoli Productivity Center for Replication session for XIV snapshots:

1. In the Tivoli Productivity Center for Replication GUI navigate to the Sessions panel and click **Create Session** (Figure 10-13).

> **Important:** The Tivoli Productivity Center for Replication GUI uses pop-up browser windows for many of the wizard-based steps. Ensure your browser allows these pop-ups to display.

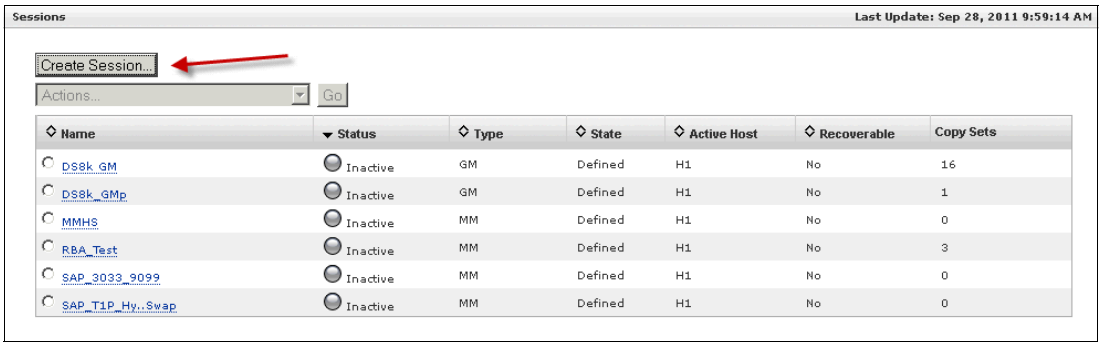


Figure 10-13 Create a new session using the Tivoli Productivity Center for Replication GUI

2. The Create Session window opens (Figure 10-14). Select the type of session to create: select **XIV** as your hardware type and click **Next**.

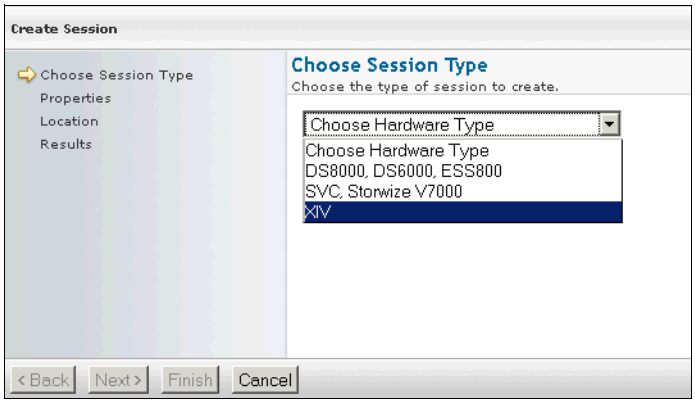


Figure 10-14 Choose Session Type

3. Choose a session type, as shown in Figure 10-15. Select **Point in Time**; **Snapshots**. Click **Next**.

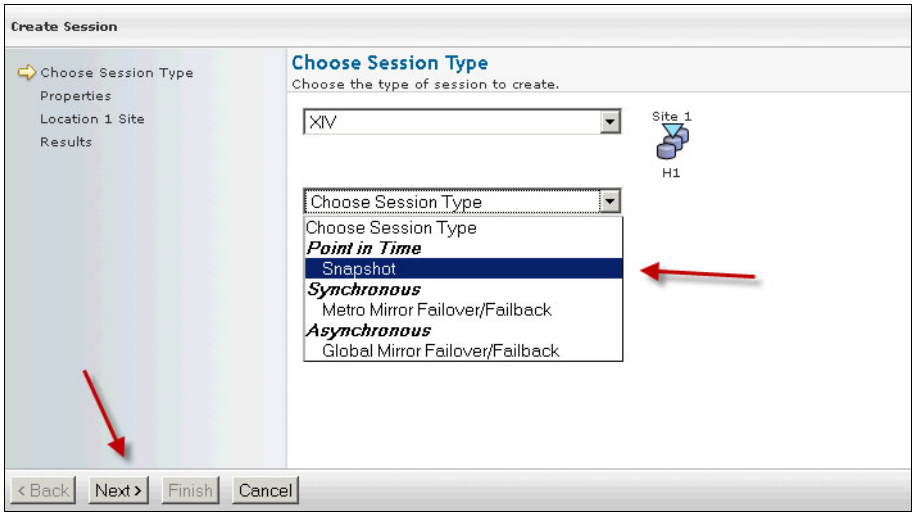


Figure 10-15 Choosing the snapshot for the new session

4. Enter the appropriate values for Session Name (required) and Description (optional) as shown in Figure 10-16. Click **Next**.

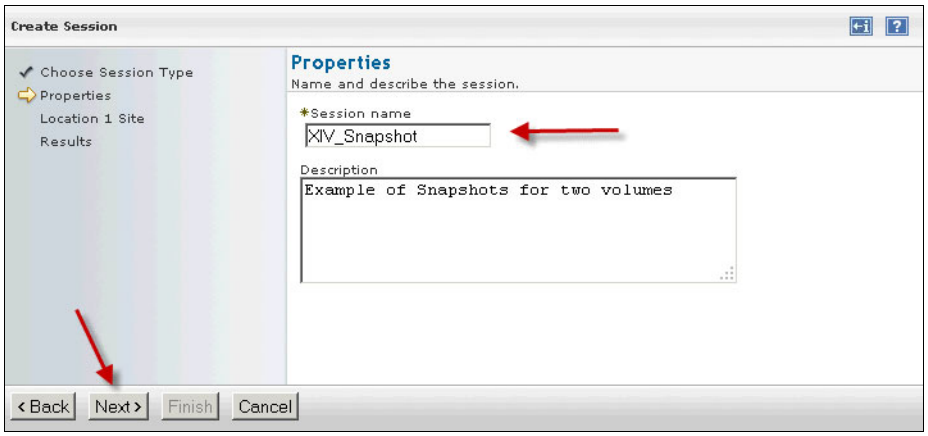


Figure 10-16 Name and description for the Snapshot session

5. As shown in Figure 10-17, Tivoli Productivity Center for Replication prompts you for a Site Location. Select the previously defined XIV storage entry from the drop-down list and click **Next**.

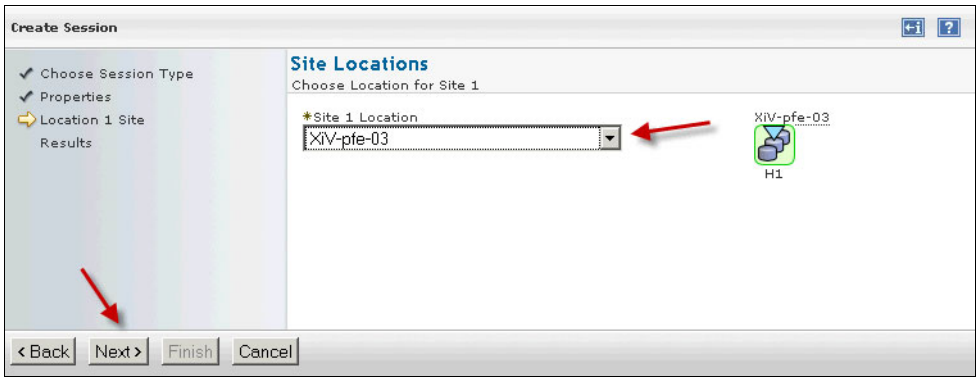


Figure 10-17 Select the XIV Host system for snapshot session

The first part of the process is completed; the session is successfully created (Figure 10-18).

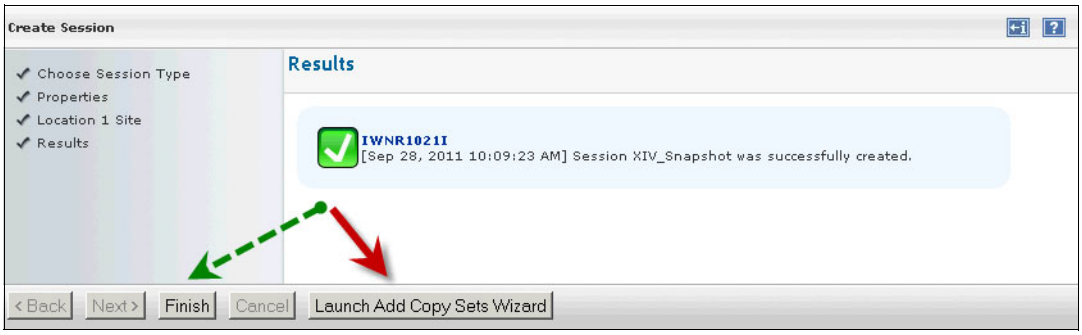


Figure 10-18 Session created successfully; Next add volumes or go to main panel

Click **Finish** to see your new, empty session (Figure 10-19).

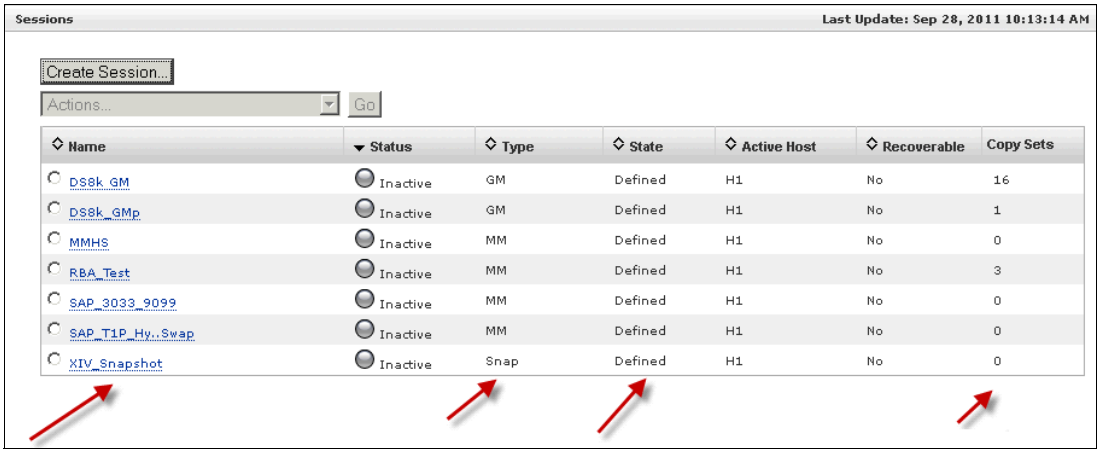


Figure 10-19 Session panel showing new session added

10.10.2 Defining and adding copy sets to a session

After a session has been defined, Tivoli Productivity Center for Replication needs to know which volumes to act on. You could have launched this process when you finished defining the Tivoli Productivity Center for Replication session if, instead of clicking Finish as in Figure 10-18 on page 347, you had clicked **Launch Add Copy Sets Wizard**.

This process can also be launched from the Tivoli Productivity Center for Replication session GUI, which is shown in Figure 10-26 on page 350.

Use either method to launch the Add Copy Sets wizard shown in Figure 10-20, then use the following steps to add copy sets:

1. Select the XIV, Pool, and Volumes that you want to add to the newly defined session for snapshots (Figure 10-20). Click **Next**.

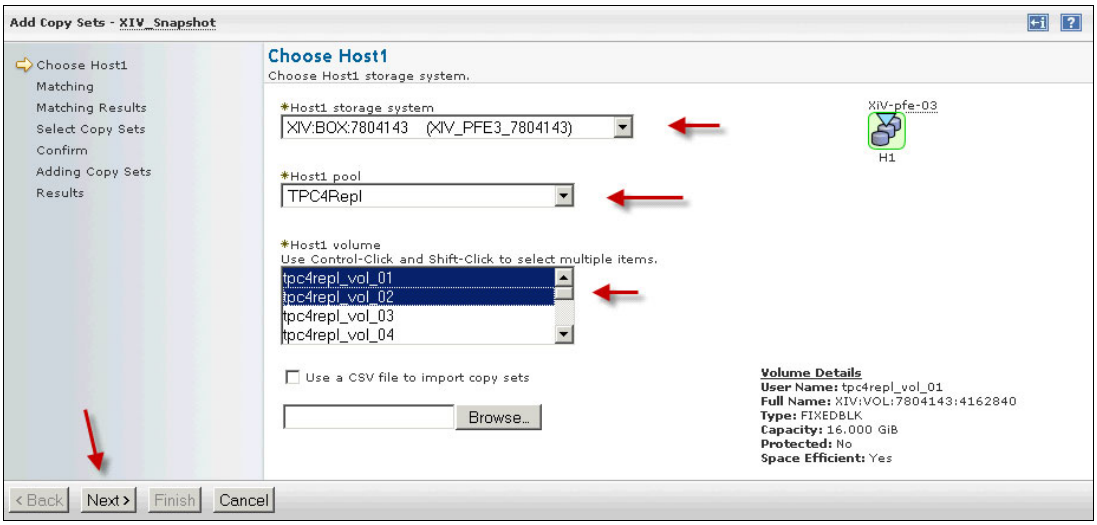


Figure 10-20 Adding a copy set to session: Choose Host, Pool, and Volumes from XIV

2. The next panel (Figure 10-21) confirms that the volumes have been added to the Tivoli Productivity Center for Replication repository. Click **Next**.

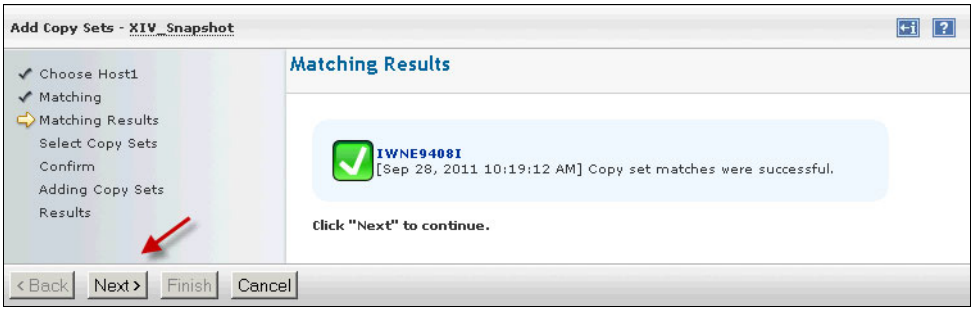


Figure 10-21 Selection of XIV volumes have been added to repository

3. Add the two volume copy set to the Tivoli Productivity Center for Replication internal repository, as shown in Figure 10-23, and click **Next**.

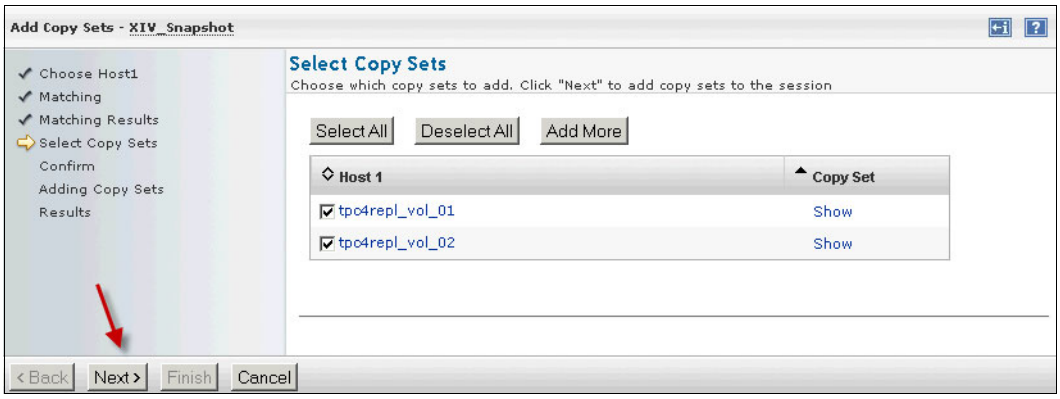


Figure 10-22 Select volumes to be added to copy sets and create consistency group

4. The confirmation panel shown in Figure 10-23 opens, indicating that Tivoli Productivity Center for Replication added the information to its repository and is now ensuring it has access to the XIV volumes. Click **Next**.

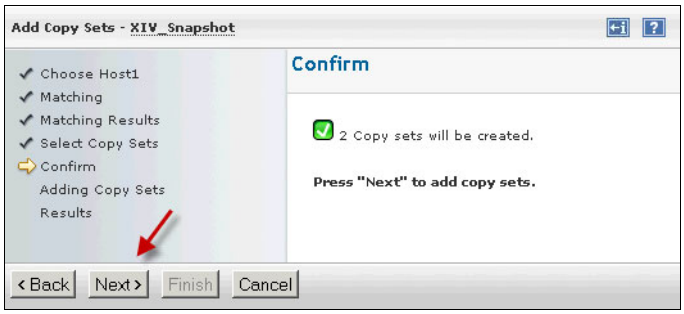


Figure 10-23 Confirming the XIV volumes will be added to consistency group

As shown in Figure 10-24, Tivoli Productivity Center for Replication completed the second phase of the process, with the successful completion of the Add Copy Sets wizard.

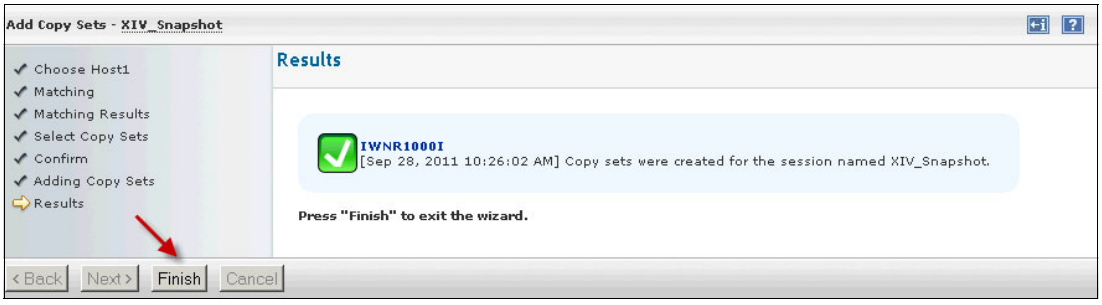


Figure 10-24 Successfully completing the XIV Snapshot copy set

Click **Finish** to display the updated Sessions window shown in Figure 10-25.

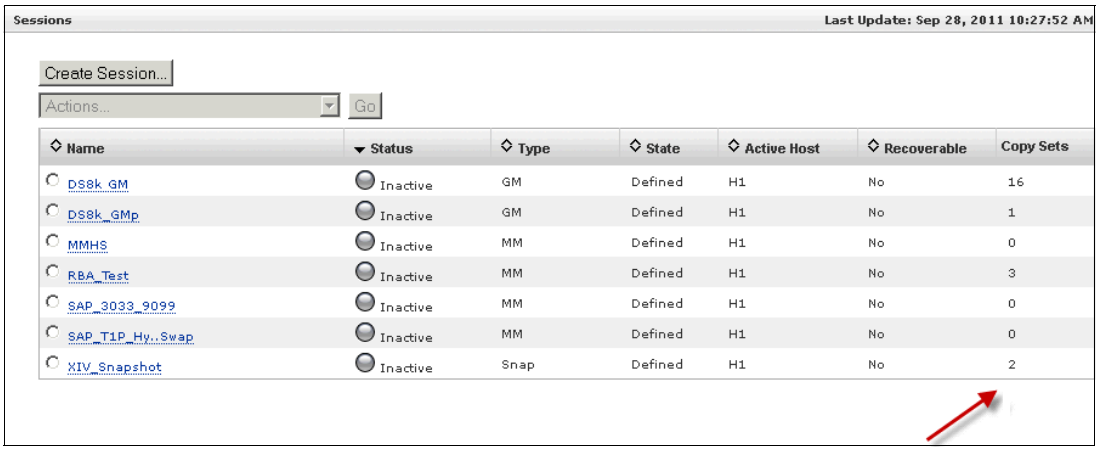


Figure 10-25 Updated Sessions panel showing the newly created snapshot session and copy set

10.10.3 Activating Snapshot session

After a session is defined, you can access its details and modify it from the Session Details window shown in Figure 10-26. Because this session has never been run, the details under Snapshot Groups remain empty.

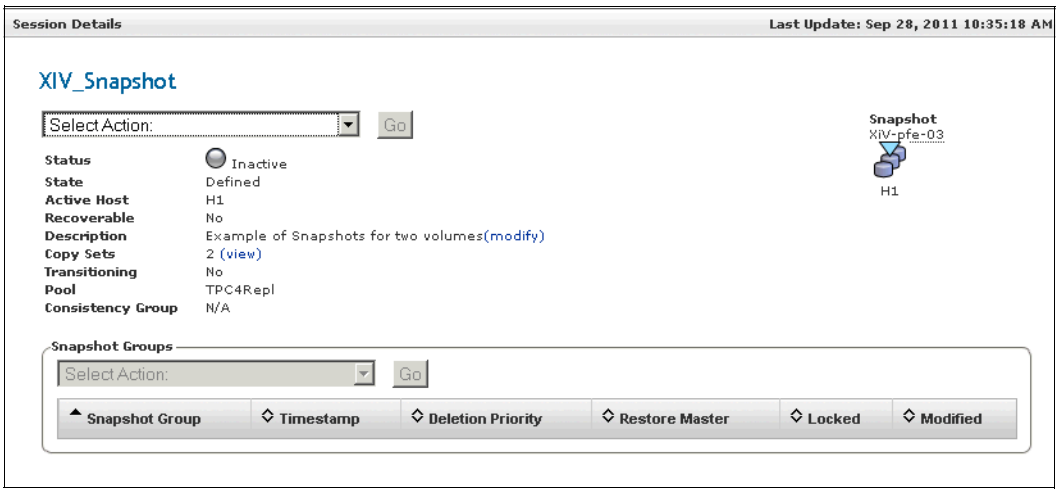


Figure 10-26 Detailed view of XIV Snapshot session; inactive and has not been run

Use the following steps to activate the Tivoli Productivity Center for Replication Snapshot session:

1. In the Session Details panel click the pull-down list, select **Create Snapshot**, and click **Go**, as shown in Figure 10-27.

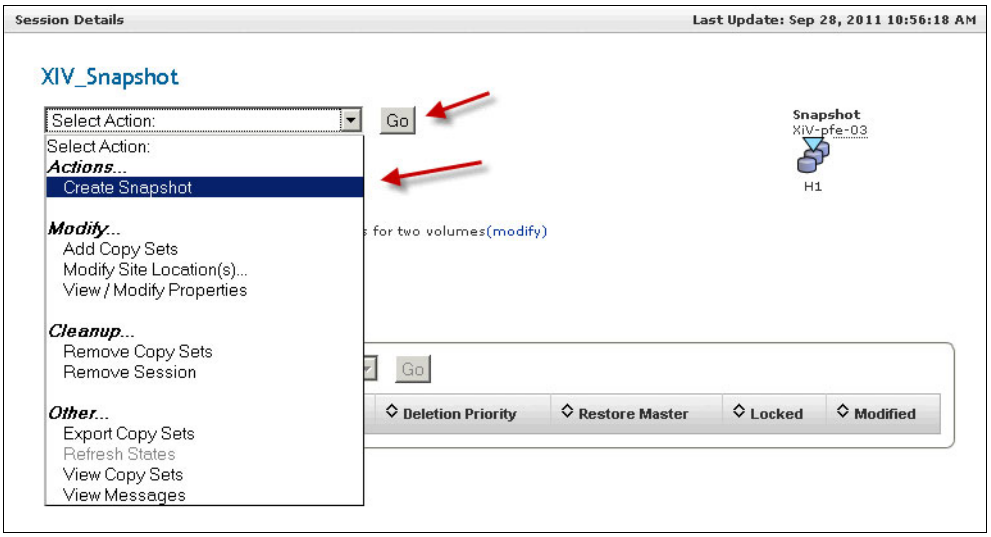


Figure 10-27 Actions available for this session; choose Create Snapshot to activate

2. A new wizard opens; it confirms the actions you are about to take on those volumes. Additionally, under Advanced Options, you can optionally modify various values specific to XIV, including the actual snapshot group name and deletion priority. This is illustrated in Figure 10-28. Make the appropriate selections and click **Yes**.

-

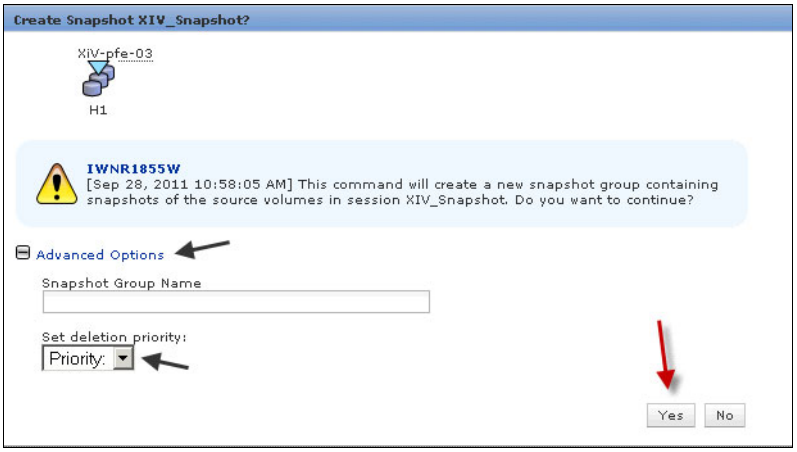

Create Snapshot XIV_Snapshot?
XiV-pfe-03
H1
IWNR1855W
[Sep 28, 2011 10:58:05 AM] This command will create a new snapshot group containing snapshots of the source volumes in session XIV_Snapshot. Do you want to continue?
Advanced Options
Snapshot Group Name
Set deletion priority:
Priority:
Yes
No

Figure 10-28 Confirmation of snapshot session activation and options

Tivoli Productivity Center for Replication now runs the snapshot command on the defined copy sets in the session. This is illustrated in Figure 10-29 on page 352. At the top portion Tivoli Productivity Center for Replication shows its actions and allows you to view more details about the operations performed, including the console. A detailed view of the Tivoli Productivity Center for Replication console log is shown in Figure 10-32 on page 353.

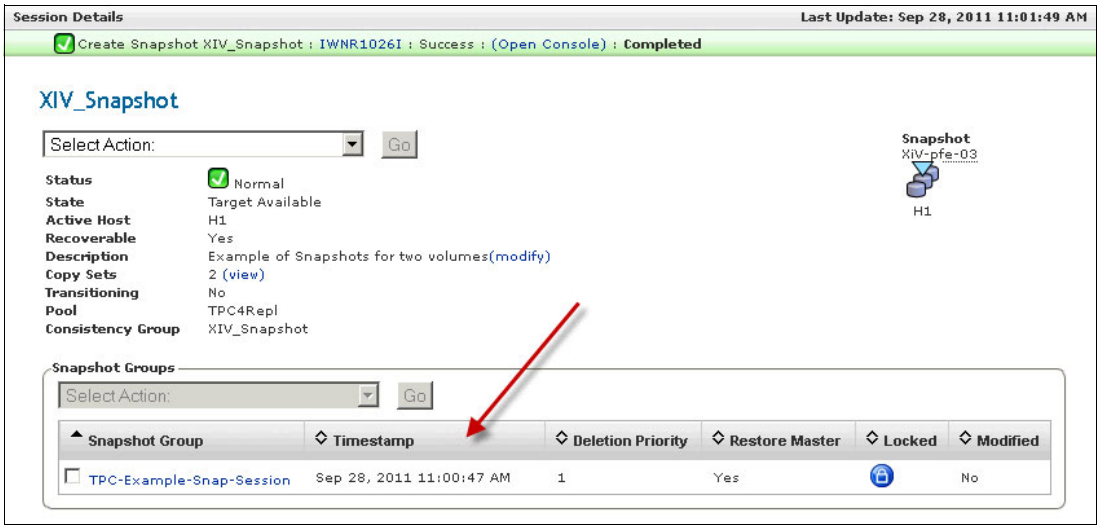


Figure 10-29 Session is activated; XIV has taken the snapshot of the volumes in the copy set

Figure 10-30, and Figure 10-31 on page 353 show, directly from the XIV GUI, the results of the recent Tivoli Productivity Center for Replication operations.

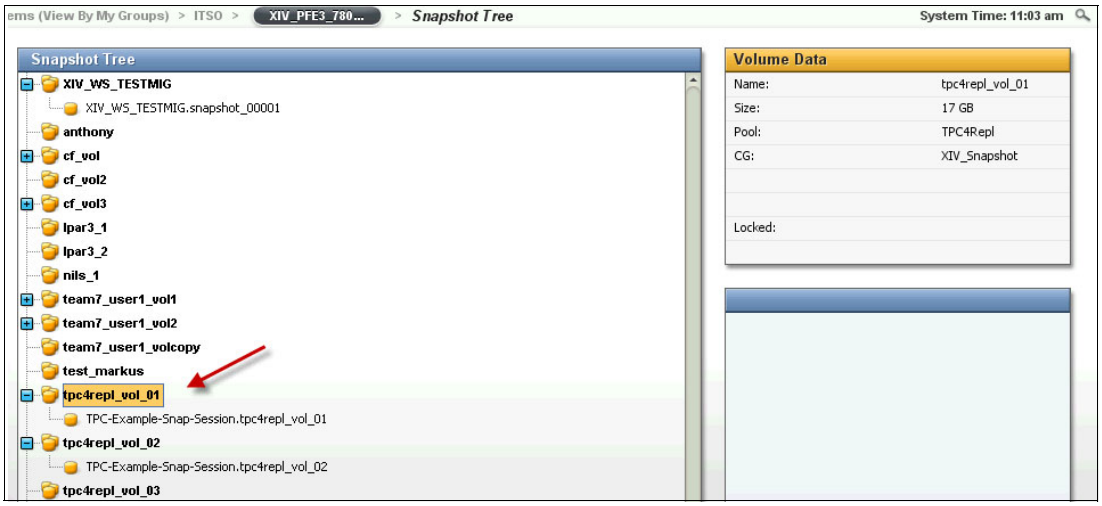


Figure 10-30 Confirmation of the Tivoli Productivity Center for Replication process; Snapshot tree view

As previously described, Tivoli Productivity Center for Replication places all copy sets in a consistency group. In Figure 10-31 on page 353, notice how Tivoli Productivity Center for Replication uses various levels inside the XIV consistency groups definitions for the single copy set created earlier.

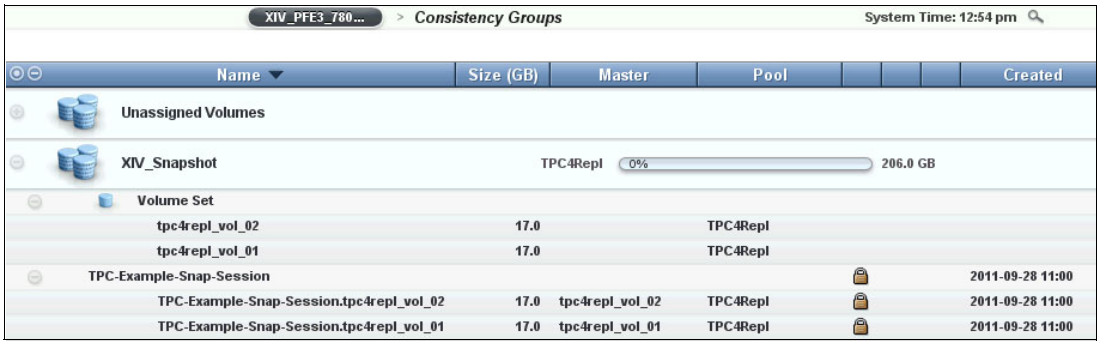


Figure 10-31 Creation of Copy Set Snapshot Session as an XIV consistency group

Tivoli Productivity Center for Replication also has a console view available, which is shown in Figure 10-32. It contains links which have parent and child relationships for many of the log entries.

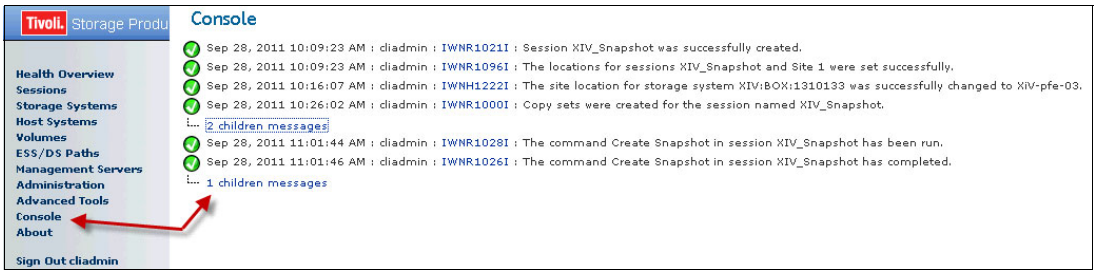

Tivoli. Storage Produ
Console
Health Overview
Sessions
Storage Systems
Host Systems
Volumes
ESS/DS Paths
Management Servers
Administration
Advanced Tools
Console
About
Sign Out cliadmin
Sep 28, 2011 10:09:23 AM : cliadmin : IWNR1021I : Session XIV_Snapshot was successfully created.
Sep 28, 2011 10:09:23 AM : cliadmin : IWNR1096I : The locations for sessions XIV_Snapshot and Site 1 were set successfully.
Sep 28, 2011 10:16:07 AM : cliadmin : IWNH1222I : The site location for storage system XIV:BOX:1310133 was successfully changed to XiV-pfe-03.
Sep 28, 2011 10:26:02 AM : cliadmin : IWNR1000I : Copy sets were created for the session named XIV_Snapshot.
2 children messages
Sep 28, 2011 11:01:44 AM : cliadmin : IWNR1028I : The command Create Snapshot in session XIV_Snapshot has been run.
Sep 28, 2011 11:01:46 AM : cliadmin : IWNR1026I : The command Create Snapshot in session XIV_Snapshot has completed.
1 children messages

Figure 10-32 Panel showing the log of actions for the entire process

10.10.4 Additional snapshot actions inside a session

After a Snapshot session has been defined, you can also take other actions, as you would from the XIV GUI or XCLI, such as a restore operation. Figure 10-33 shows several possible actions.

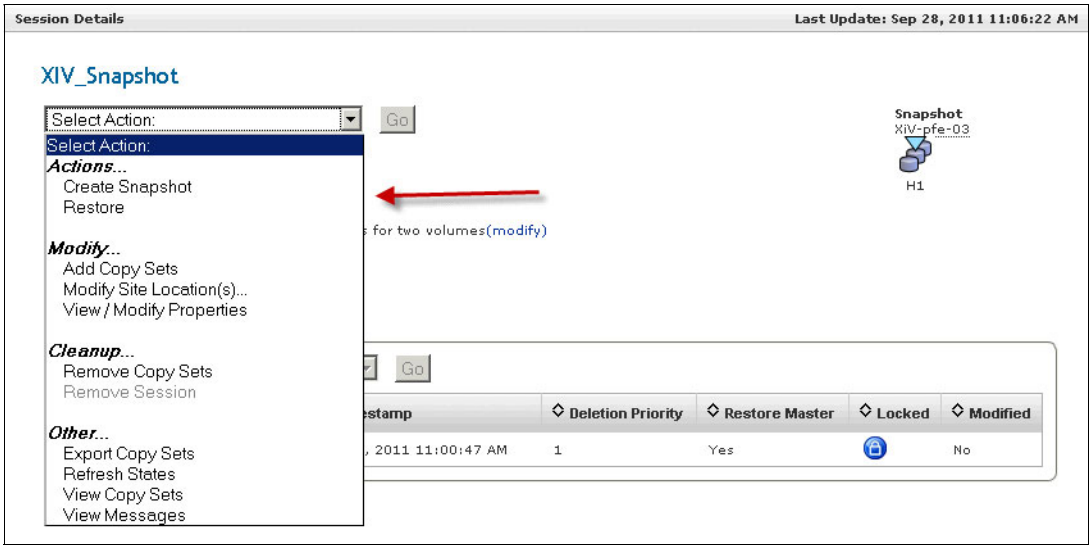


Figure 10-33 Panel showing some available actions

10.11 XIV synchronous mirroring (Metro Mirror)

As for Snapshots, to set up Metro Mirror, you must create and define a new session for this type of Copy Service, add the XIV volumes from both XIVs to this newly defined session, and finally, activate the session.

Both the XIV pool and volumes must be defined using the XIV GUI or XCLI before using Tivoli Productivity Center for Replication for this process.

10.11.1 Defining a session for Metro Mirror

Use the following steps to define a Tivoli Productivity Center for Replication session for Metro Mirror:

1. In the Tivoli Productivity Center for Replication GUI navigate to the Sessions window and click **Create Session**.

   A comparable process was shown in detail previously, beginning with Figure 10-13 on page 345. The session creation process is similar across all session types, so not all panels are repeated here.

2. Select the XIV as the storage type and click **Next**.
3. Define the Tivoli Productivity Center for Replication session type, as shown in Figure 10-34. Circled in red in the upper right section of the panel is an icon that changes according to the session types. The session wizard also varies slightly depending on the session type being defined.

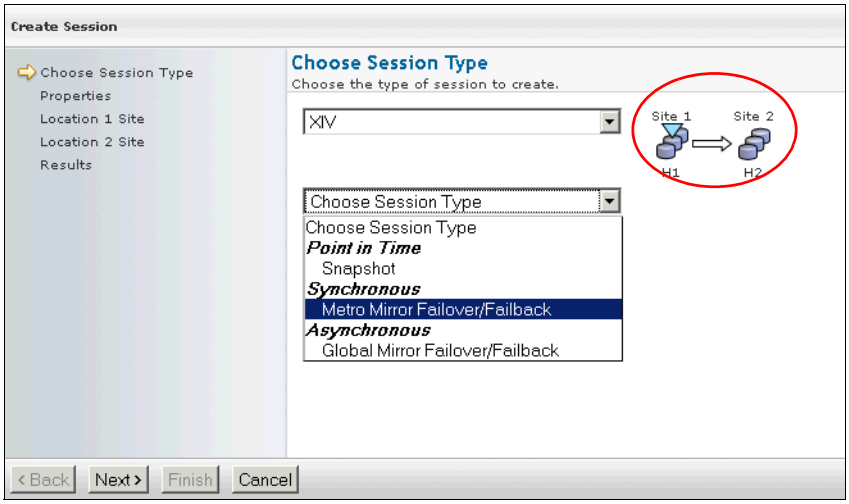


Figure 10-34 Session wizard; Synchronous Metro Mirror option

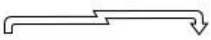
Select **Synchronous** → **Metro Mirror**, and click **Next** to proceed with the definition of the session properties.

Table 10-1 on page 355 and Table 10-2 on page 355 explain the session images. The default site names (Site1 and Site2) can be customized; H1 and H2 are Host 1 and Host 2, and cannot be changed.

Table 10-1 Volume role symbols

Symbol	Description	Meaning
	Active host volumes	This symbol represents volumes that contain the source of updated tracks to which the application is actively issuing read and write input/output (I/O).
	Recoverable volumes	This symbol represents volumes that contain a consistent copy of the data.
	Inconsistent volumes	This symbol represents the volumes that do not contain a consistent copy of the data.
	Selected volumes	This symbol represents the volumes that are selected for an operation (for example, changing location or displaying role pair information).

Table 10-2 Data copying symbols

Symbol	Description	Meaning
	Metro Mirror copying	This symbol represents a copying Metro Mirror relationship.
	Metro Mirror copying with errors	This symbol represents a Metro Mirror relationship that is copying, but with errors on one or more pairs.
	Metro Mirror inactive	This symbol represents an inactive Metro Mirror relationship.
	Metro Mirror inactive with errors	This symbol represents an inactive Metro Mirror relationship with errors on one or more pairs.
	Global Copy copying	This symbol represents a copying Global Copy relationship.
	Global Copy copying with errors	This symbol represents a copying Global Copy relationship with errors on one or more pairs.
	Global Copy inactive	This symbol represents an inactive Global Copy relationship.
		
	Global Copy inactive with errors	This symbol represents an inactive Global Copy relationship with errors on one or more pairs.

4. In the Properties panel shown in Figure 10-35 enter a Session name (required) and a Description (optional) and click **Next**.

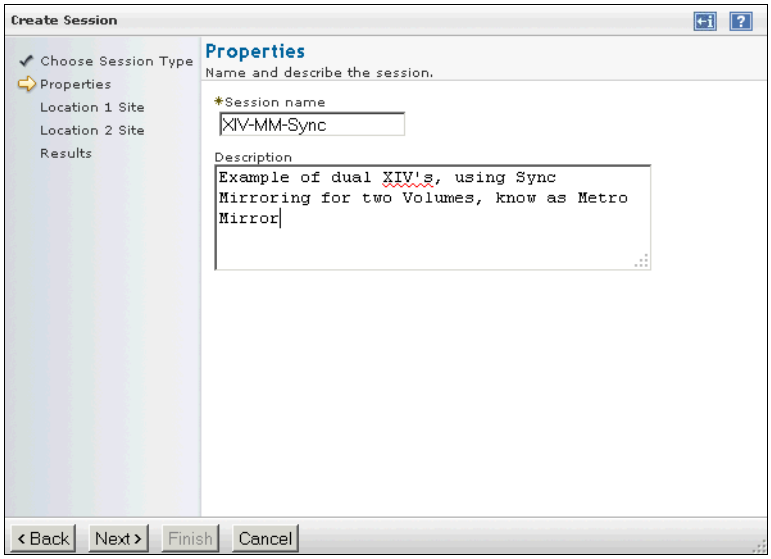


Figure 10-35 Session wizard; enter descriptive name and metadata

5. In the Site Locations panel shown in Figure 10-36, from the Site 1 Location pull-down list, select the site of the first XIV. The pull-down list shows the various sites defined to Tivoli Productivity Center for Replication. Click **Next**.

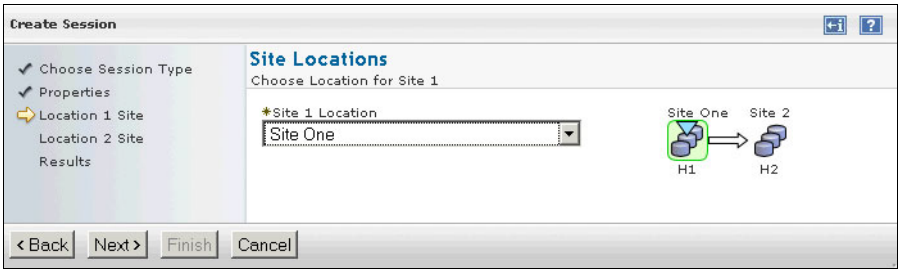


Figure 10-36 Session wizard, synchronous copy set pair

6. Define the secondary or target site, as shown in Figure 10-37. From the Site 2 Location pull-down list, select the appropriate target or secondary site that has the target XIV and corresponding volumes. Click **Next**.

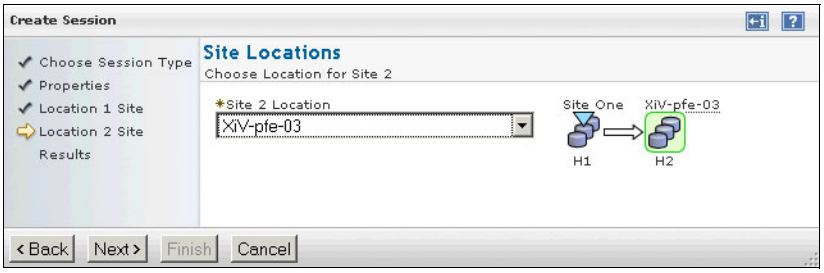


Figure 10-37 Session wizard showing site location for secondary site

Tivoli Productivity Center for Replication creates the session and displays the result as shown in Figure 10-38.

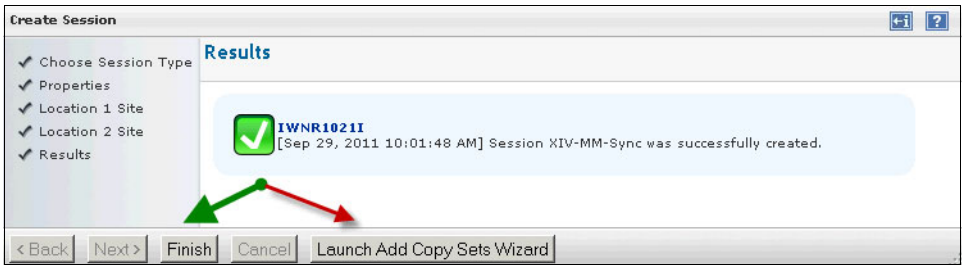
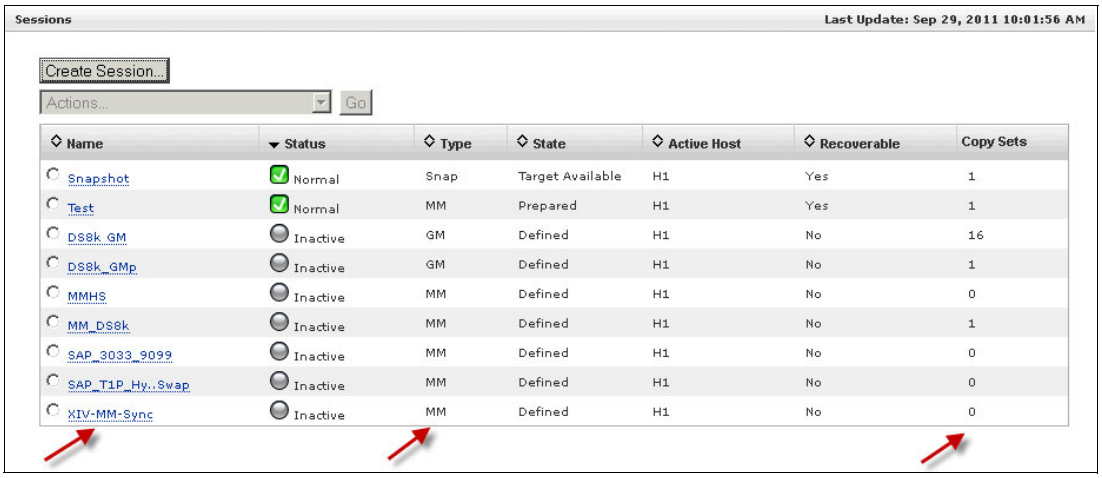


Figure 10-38 Tivoli Productivity Center for Replication Wizard completes the new Metro Mirror Session.

You can click **Finish** to view the session, as shown in Figure 10-39.



Name	Status	Type	State	Active Host	Recoverable	Copy Sets
Snapshot	Normal	Snap	Target Available	H1	Yes	1
Test	Normal	MM	Prepared	H1	Yes	1
DS8k_GM	Inactive	GM	Defined	H1	No	16
DS8k_GMp	Inactive	GM	Defined	H1	No	1
MMHS	Inactive	MM	Defined	H1	No	0
MM_DS8k	Inactive	MM	Defined	H1	No	1
SAP_3033_9099	Inactive	MM	Defined	H1	No	0
SAP_T1P_Hy..Swap	Inactive	MM	Defined	H1	No	0
XIV-MM-Sync	Inactive	MM	Defined	H1	No	0

Figure 10-39 New Metro Mirror Session without any copy sets

10.11.2 Defining, adding copy sets to a Metro Mirror session

After a Metro Mirror session has been defined, the XIV volumes must be specified for that session. The following steps are used to do this.

1. The Copy Sets wizard will have various dependent drop-down menus as shown in Figure 10-40. Select the XIV in the first site, the pool, and the first volume you would like to have as part of the Metro Mirror copy set. Click **Next**.

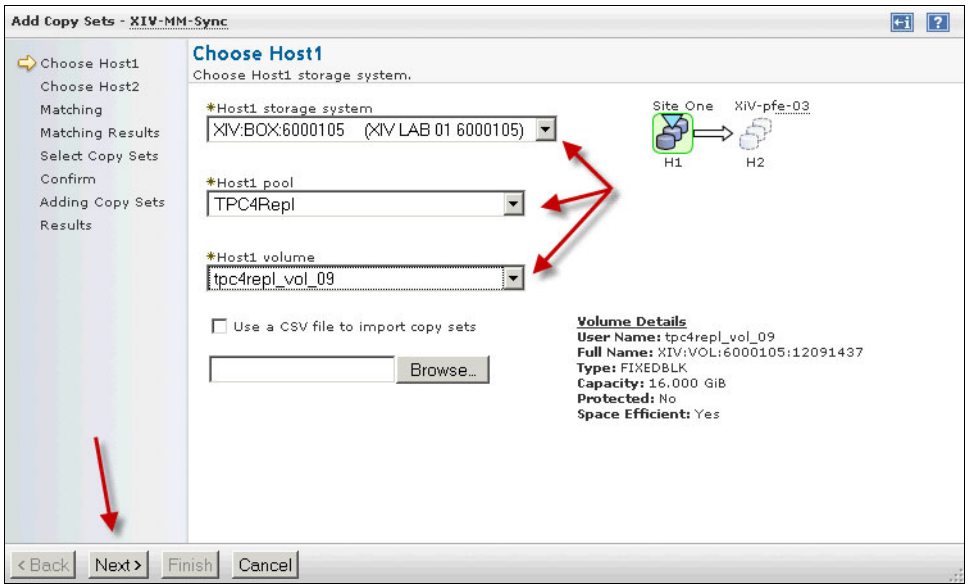


Figure 10-40 Initial step of Tivoli Productivity Center for Replication Add Copy Sets wizard for Metro Mirror

2. Make the appropriate selections for site 2 (target site), as shown in Figure 10-41. Click **Next**.

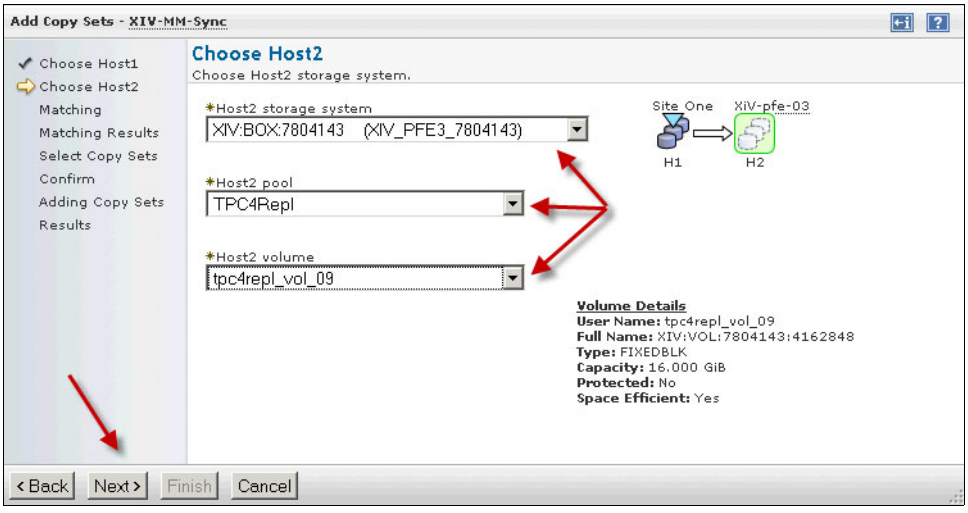


Figure 10-41 Target panel for the first volume of the Add Copy Sets wizard

3. Add more volumes to the copy sets. Figure 10-42 shows the first volume defined for the copy set.

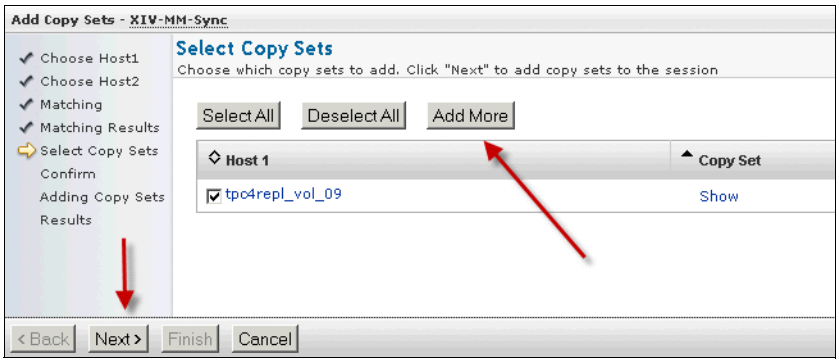


Figure 10-42 Confirming first volume selection for this copy set

4. You can add a second volume to the copy set. Depending on your business needs you might have several volumes (all within the same pool at each XIV) in one copy set, or individual volumes.

   To add another volume, click **Add More**. Tivoli Productivity Center for Replication keeps track of the first volume, as you will see when you complete the wizard.

5. Figure 10-43 shows the second volume, (vol_10) that we are adding to the copy set (we have also selected the same values for the primary XIV and pool). Make the appropriate selections and click **Next**.

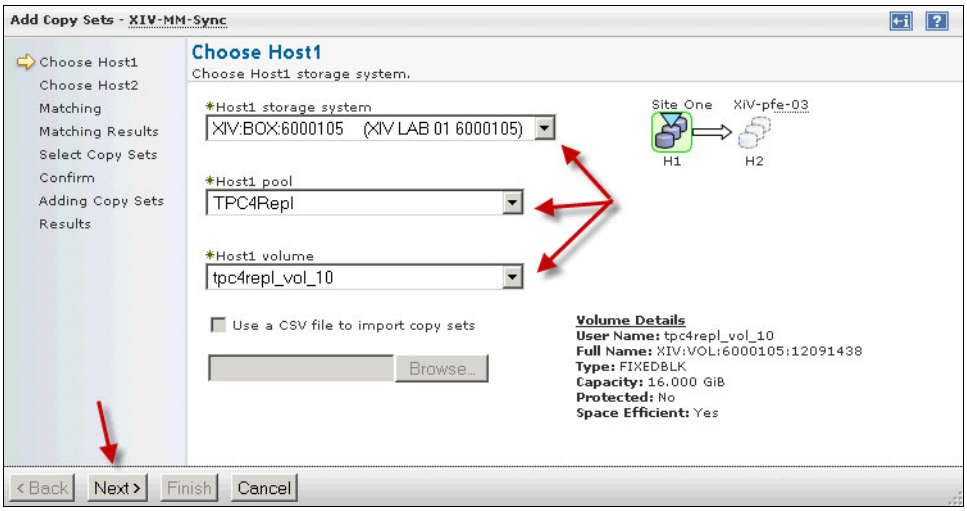


Figure 10-43 Adding a second volume to copy set; similar menus as previous

6. The wizard prompts for the secondary XIV values as shown in Figure 10-44. Make the appropriate entries and click **Next**.

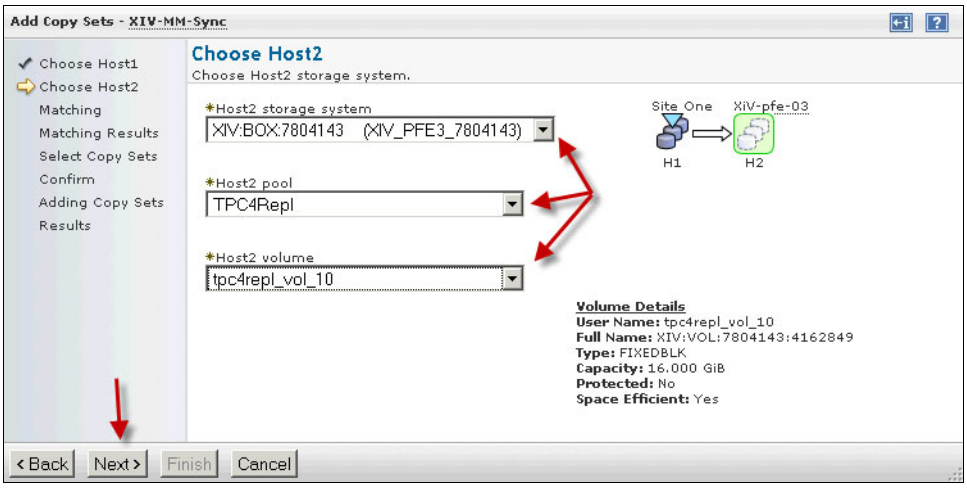


Figure 10-44 Add Copy Sets wizard for second XIV and target volume selection panel

The Copy Set wizard now has both volumes selected, and you have the option to add more volumes, if required.

Note: If you need to add a large set of volumes, you have the option to import the volume definitions and pairings from a comma-separated variable (.csv) file. See the Tivoli Productivity Center Information Center for details and examples:

http://pic.dhe.ibm.com/infocenter/tivihelp/v59r1/index.jsp

7. Figure 10-45 shows the confirmation panel that is returned. Click **Next**.

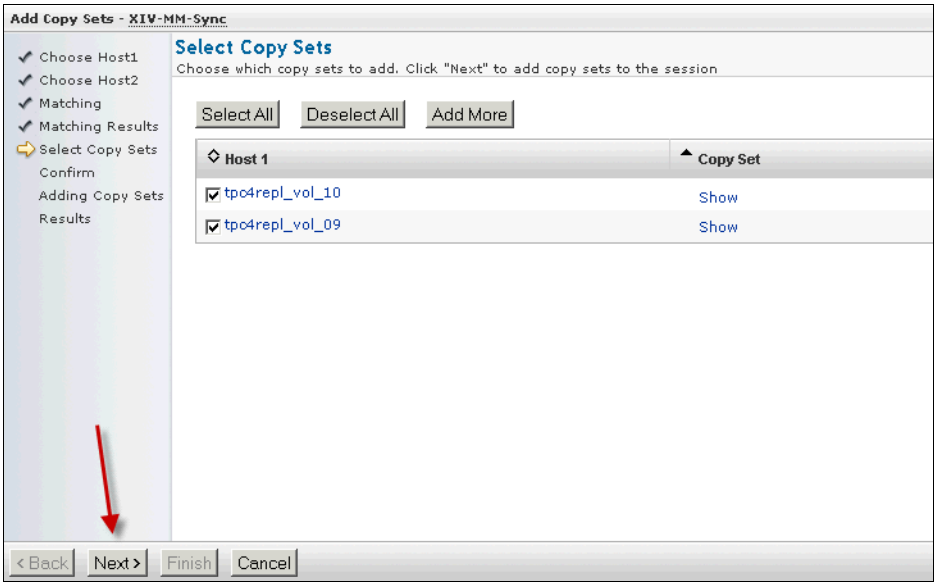


Figure 10-45 Copy set confirmation panel showing both volumes added

8. Tivoli Productivity Center for Replication confirms the volumes being added to the set, as shown in Figure 10-46. Click **Next**.

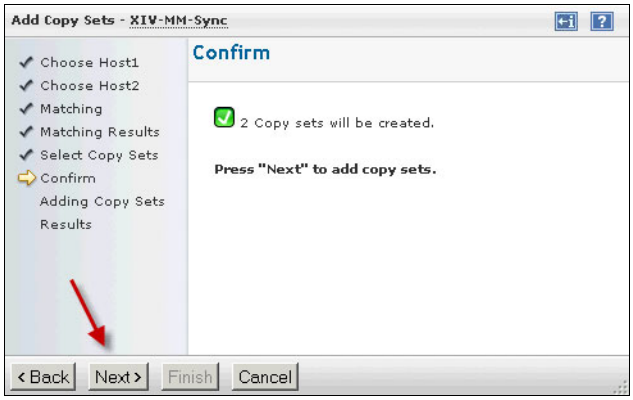


Figure 10-46 Add Copy Sets wizard asking to confirm the addition of both volumes to set

Tivoli Productivity Center for Replication updates its repository, indicating progress of the update as shown in Figure 10-47.

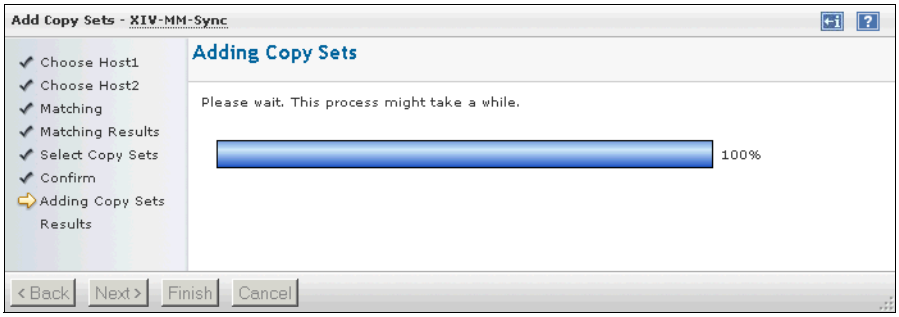


Figure 10-47 Add Copy Sets wizard in progress of updating repository

After Tivoli Productivity Center for Replication completes the Copy Set operation, the Results panel is displayed as shown in Figure 10-48.

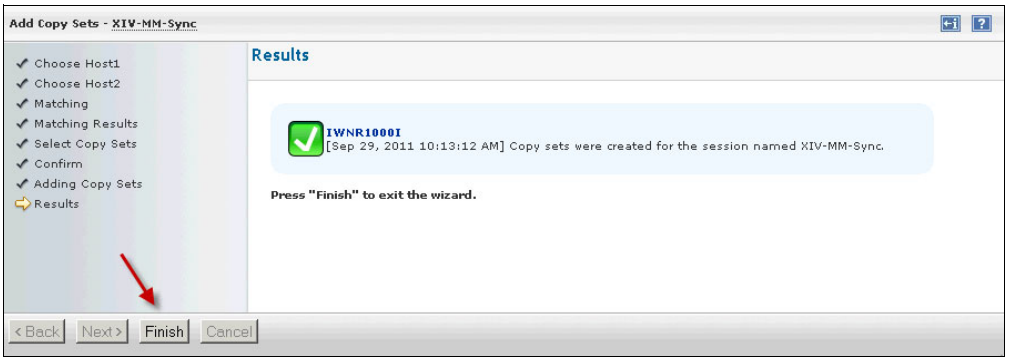


Figure 10-48 Copy Set Wizard for Metro Mirror session completion

9. Click **Finish** to view the updated Session Details window as shown in Figure 10-49.

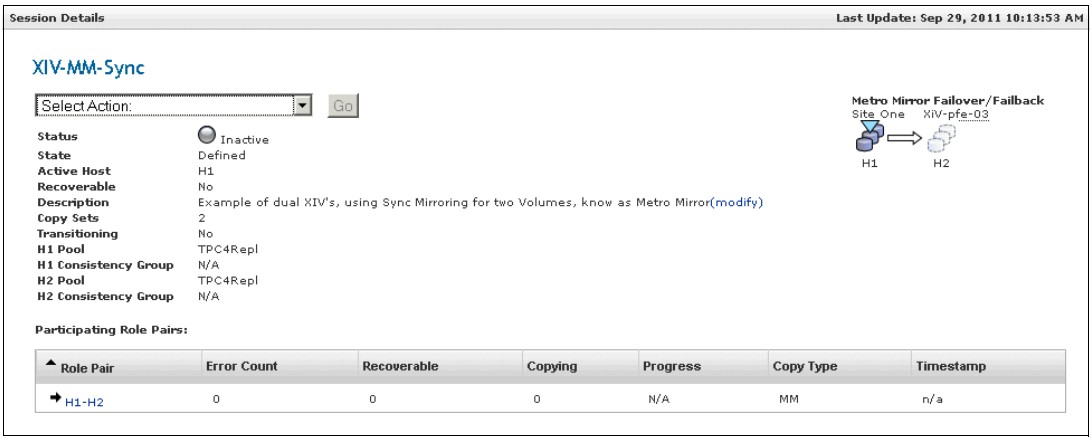


Figure 10-49 Metro Mirror session details at the completion of both wizards

After defining a session, you can access its details and modify it from the Session Details window depicted in Figure 10-49.

10.11.3 Activating a Metro Mirror session

Now that you have defined a session and added a copy set containing volumes, you can move on to the next phase, and activate the session using the following steps:

1. From the Select Action menu, select **Start H1->H2**, and click **Go** to activate the session, as shown in Figure 10-50.

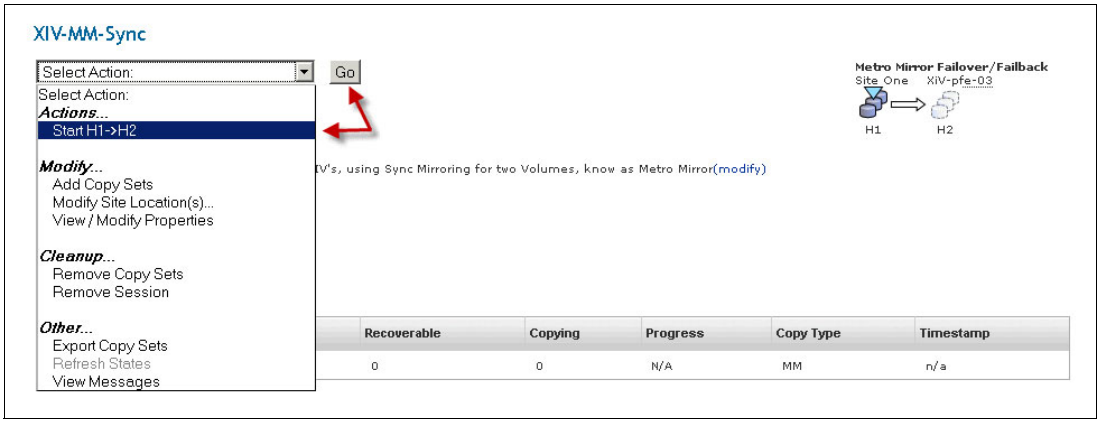


Figure 10-50 Action items available to the Metro Mirror session

2. A window prompts for confirmation (Figure 10-51). Click **Yes** to confirm.

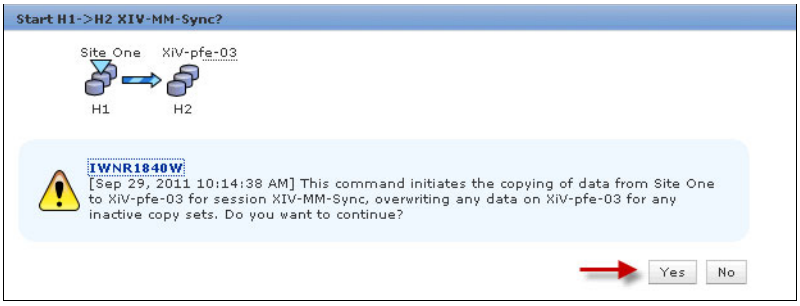

Start H1->H2 XIV-MM-Sync?
Site One
XiV-pfe-03
H1
H2
IWNR1840W
[Sep 29, 2011 10:14:38 AM] This command initiates the copying of data from Site One to XiV-pfe-03 for session XIV-MM-Sync, overwriting any data on XiV-pfe-03 for any inactive copy sets. Do you want to continue?
Yes
No

Figure 10-51 Last warning before taking the Metro Mirror session active

Tivoli Productivity Center for Replication communicates with the XIV systems to activate the session and displays an updated Session Details window as illustrated in Figure 10-52.

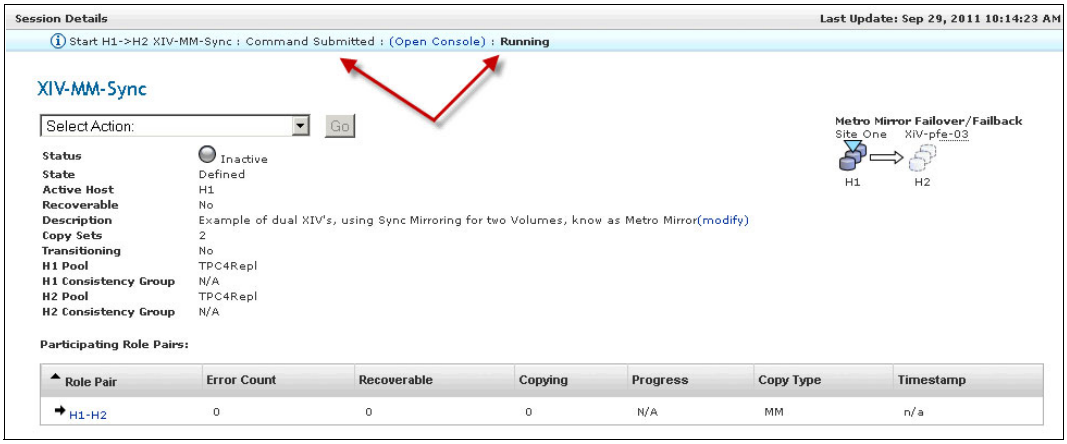


Figure 10-52 Activating the session

After the Tivoli Productivity Center for Replication commands are sent to the XIV, Tivoli Productivity Center for Replication continues to update the same Session Details window to reflect the latest status (Figure 10-53).

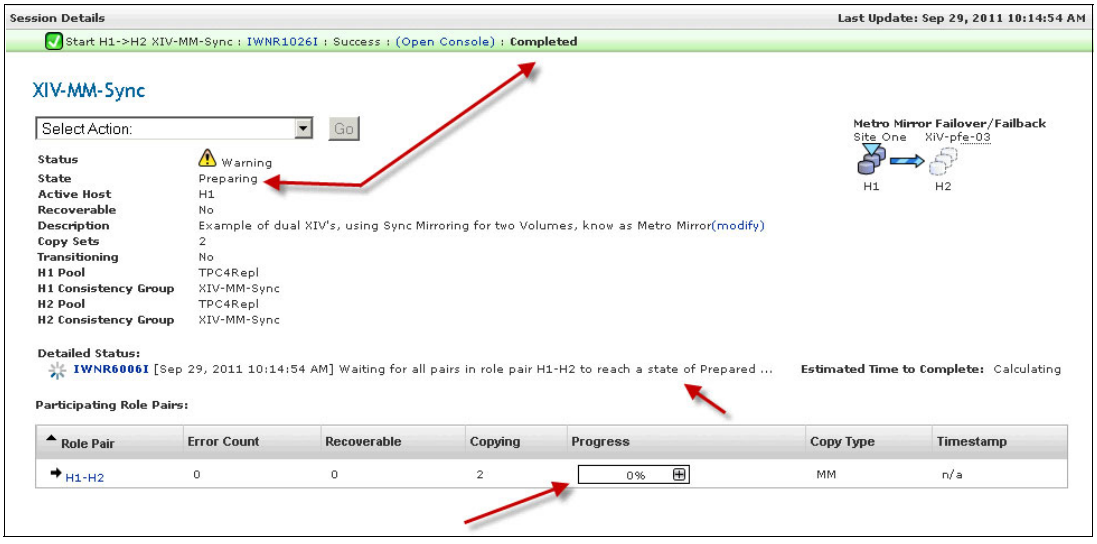


Figure 10-53 Session Details panel: Various progress actions for Metro Mirror session

Figure 10-54 shows the corresponding session information from the XIV GUI.

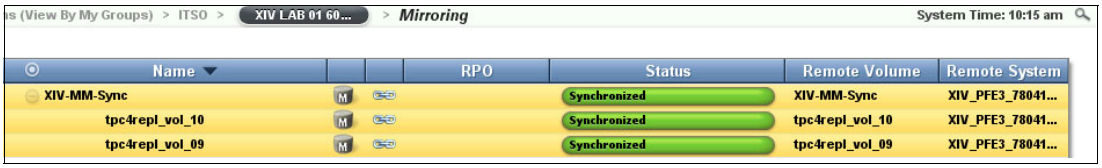


Figure 10-54 XIV GUI showing the same information as inside Tivoli Productivity Center for Replication

10.11.4 Suspending the Metro Mirror (XIV Synchronous Mirror) session

At some point you might want to suspend the Metro Mirror session. This might be dictated by various business reasons, or physical issues such as a communication link failure, or a true site outage. Suspending the Mirror session is also the first step for allowing the target volumes to become accessible, and also reversing the actual mirror direction.

In any case, to make the target volumes available, you must access the Session, and perform a Suspend and then Recover. This procedure is accomplished in the following steps:

1. Navigate to the Session Details panel and select **Suspend**, as shown in Figure 10-55.

Figure 10-55 Available actions for a Metro Mirror Session, including Suspend

The updated Session Details window, as a result of the Suspend action, is shown in Figure 10-56.

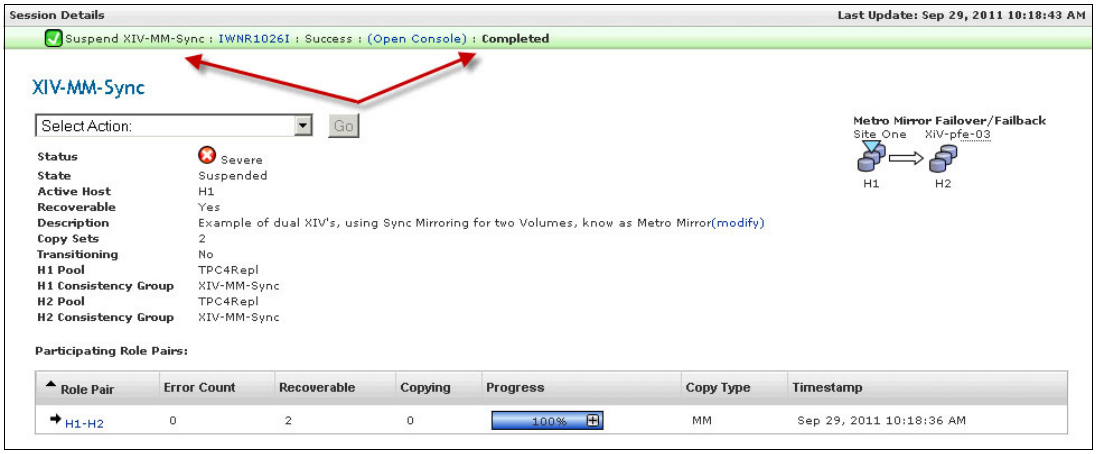


Figure 10-56 Tivoli Productivity Center for Replication Metro Mirror Session being suspended

Figure 10-57 shows the same status observed directly from the XIV GUI.

Figure 10-57 XIV GUI showing the same information for suspended Metro Mirror Session

Alternatively, you can look at the Tivoli Productivity Center for Replication Console log for a list of all of the actions taken, as illustrated in Figure 10-58.

Console
Last Update: Sep 29, 2011 11:01:44 AM
Sep 29, 2011 9:35:11 AM : cliadmin : IWNR1021I : Session XIV-MM-Sync-Session was successfully created.
Sep 29, 2011 9:35:12 AM : cliadmin : IWNR1096I : The locations for sessions XIV-MM-Sync-Session and Site 1 were set successfully.
Sep 29, 2011 9:35:12 AM : cliadmin : IWNR1096I : The locations for sessions XIV-MM-Sync-Session and Site 2 were set successfully.
Sep 29, 2011 9:35:12 AM : cliadmin : IWNR1228I : The options for session XIV-MM-Sync-Session have been set successfully.
Sep 29, 2011 9:38:45 AM : cliadmin : IWNR1022I : Session XIV-MM-Sync-Session was successfully deleted.
Sep 29, 2011 9:47:30 AM : Server : IWNH0006I : The IBM Tivoli Storage Productivity Center for Replication server has been initialized with 92 volumes for storage system XIV:BOX:1310133.
Sep 29, 2011 9:56:50 AM : cliadmin : IWNR1021I : Session XIV-MM-Sync was successfully created.
Sep 29, 2011 9:56:50 AM : cliadmin : IWNR1096I : The locations for sessions XIV-MM-Sync and Site 1 were set successfully.
Sep 29, 2011 9:56:50 AM : cliadmin : IWNR1096I : The locations for sessions XIV-MM-Sync and Site 2 were set successfully.
Sep 29, 2011 9:56:50 AM : cliadmin : IWNR1228I : The options for session XIV-MM-Sync have been set successfully.
Sep 29, 2011 9:57:54 AM : cliadmin : IWNR1022I : Session XIV-MM-Sync was successfully deleted.
Sep 29, 2011 10:01:48 AM : cliadmin : IWNR1021I : Session XIV-MM-Sync was successfully created.
Sep 29, 2011 10:01:48 AM : cliadmin : IWNR1096I : The locations for sessions XIV-MM-Sync and Site 1 were set successfully.
Sep 29, 2011 10:01:49 AM : cliadmin : IWNR1096I : The locations for sessions XIV-MM-Sync and Site 2 were set successfully.
Sep 29, 2011 10:01:49 AM : cliadmin : IWNR1228I : The options for session XIV-MM-Sync have been set successfully.
Sep 29, 2011 10:14:49 AM : cliadmin : IWNR1028I : The command Start H1->H2 in session XIV-MM-Sync has been run.
Sep 29, 2011 10:14:49 AM : cliadmin : IWNR6000I : Starting all pairs in role pair H1-H2 ...
Sep 29, 2011 10:14:53 AM : cliadmin : IWNR6006I : Waiting for all pairs in role pair H1-H2 to reach a state of Prepared ...
Sep 29, 2011 10:14:53 AM : cliadmin : IWNR1026I : The command Start H1->H2 in session XIV-MM-Sync has completed.
2 children messages
Sep 29, 2011 10:14:53 AM : Server : IWNR1950I : Session XIV-MM-Sync changed from the Defined state to the Preparing state.
Sep 29, 2011 10:15:26 AM : Server : IWNR1950I : Session XIV-MM-Sync changed from the Preparing state to the Prepared state.
Sep 29, 2011 10:18:36 AM : cliadmin : IWNR1028I : The command Suspend in session XIV-MM-Sync has been run.
Sep 29, 2011 10:18:36 AM : cliadmin : IWNR6002I : Suspending all pairs in role pair H1-H2 ...
Sep 29, 2011 10:18:36 AM : cliadmin : IWNR1026I : The command Suspend in session XIV-MM-Sync has completed.
Sep 29, 2011 10:18:36 AM : Server : IWNR1950I : Session XIV-MM-Sync changed from the Prepared state to the Suspended state.

Figure 10-58 Tivoli Productivity Center for Replication Console log of Metro Mirror Session details

2. After you suspend a Metro Mirror link you can perform a recover operation, which causes Tivoli Productivity Center for Replication to reverse the link, and begin to move information from the target/slave volume back to the master/primary volume.

   This is also known as moving data from the secondary back to the primary. Tivoli Productivity Center for Replication can do this operation only after the link is suspended.

Notice the difference in the Session Details window shown in Figure 10-59 and the window in Figure 10-55 on page 364. Because the link was suspended, Tivoli Productivity Center for Replication now allows a recover operation. Select **Recover** and click **Go**.

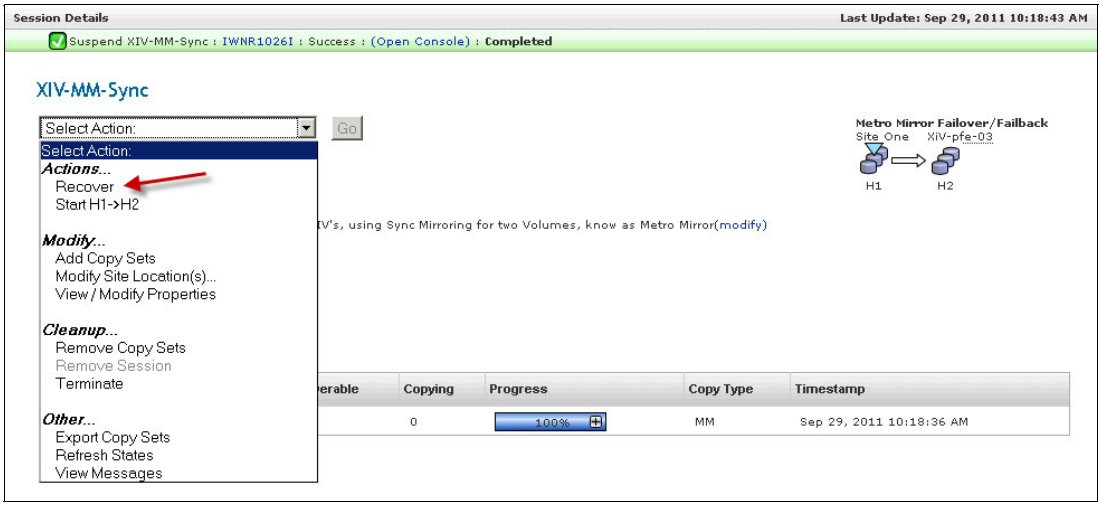


Figure 10-59 Session Details panel showing recover option available

3. Tivoli Productivity Center for Replication prompts you to confirm the operation (Figure 10-60). Click **Yes**.

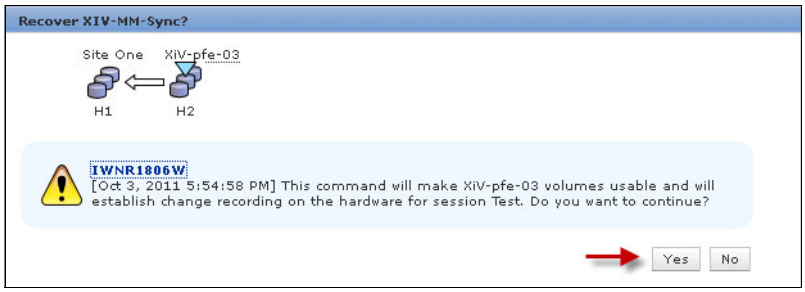

Recover XIV-MM-Sync?
Site One
XiV-pfe-03
H1
H2
IWNR1806W
[Oct 3, 2011 5:54:58 PM] This command will make XiV-pfe-03 volumes usable and will establish change recording on the hardware for session Test. Do you want to continue?
Yes
No

Figure 10-60 Final confirmation before reversing link for Metro Mirror Session

Tivoli Productivity Center for Replication now prepares both XIVs for the upcoming role change. This causes the target volumes to be available immediately.

4. You also have the option of replacing and updating the Primary/Master volume with information from the Target/Slave volume (Production Site Switch). From the Select Action list, a new choice is available, **Enable Copy to Site 1** (Figure 10-61 on page 367).

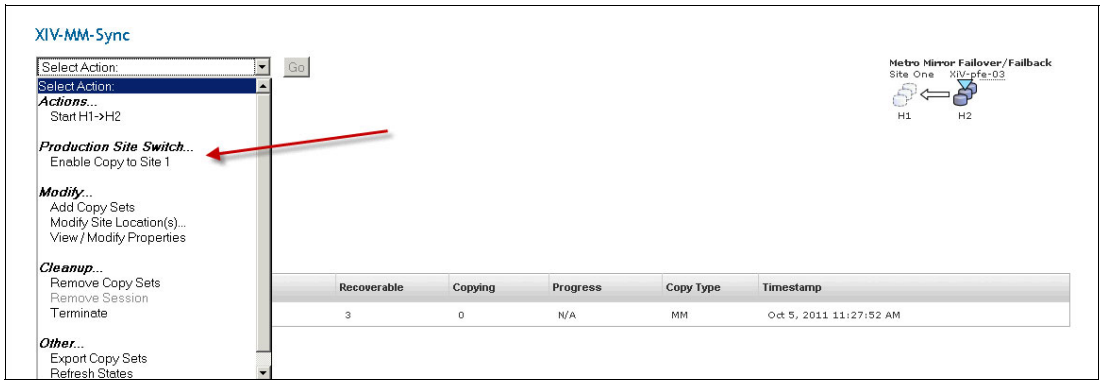


Figure 10-61 Tivoli Productivity Center for Replication Metro Mirror preparing to reverse link

The icon has the blue triangle over H2, indicating that the mirror session has switched and Site 2 is now active.

5. Click **Go** and then confirm the selection, which causes Tivoli Productivity Center for Replication to send the appropriate commands to both XIVs.
6. Activate the reversal, as shown in Figure 10-62.

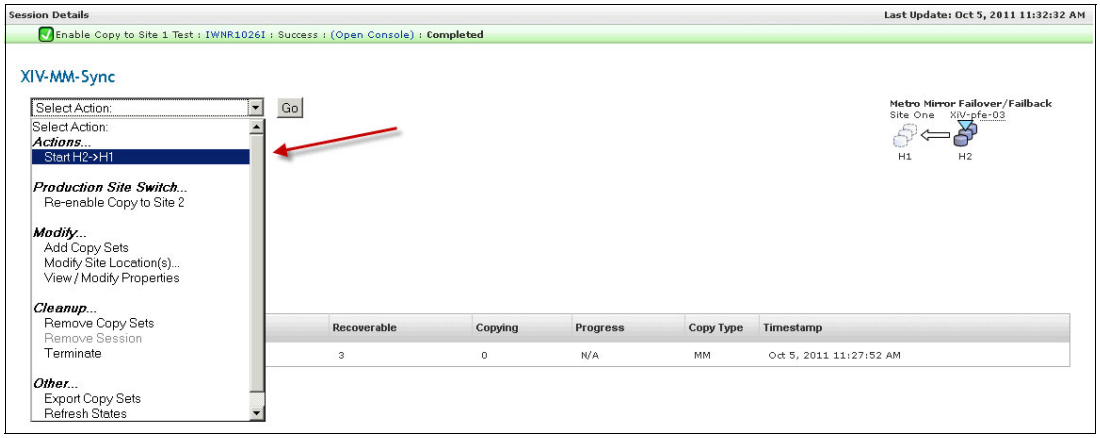


Figure 10-62 Metro Mirror before activating the link in the reverse direction

7. After the reversal, you must activate the link, shown as the **Start H2->H1** menu choice now available in the drop-down list in Figure 10-62. Click **Go** and confirm to have Tivoli Productivity Center for Replication activate the link in reverse.

Figure 10-63 on page 368 shows that Tivoli Productivity Center for Replication has activated the link in reverse, and the volumes have fully replicated themselves back to the original source volumes.

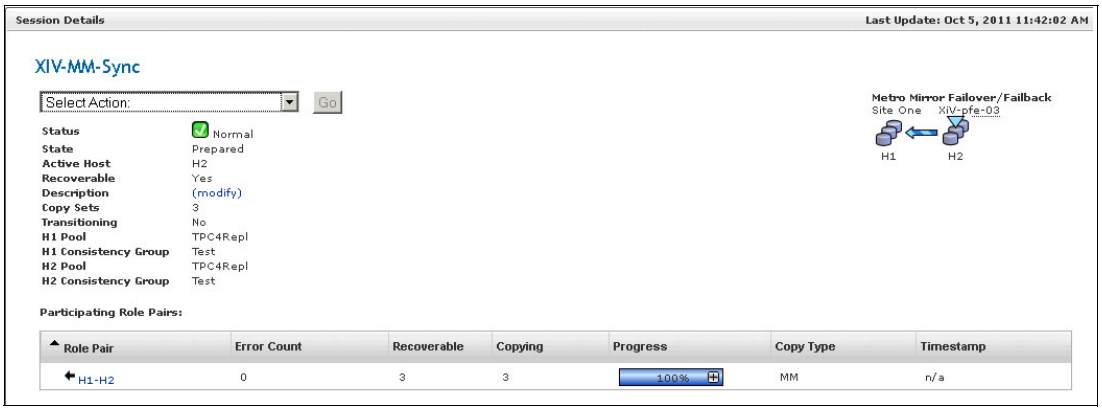


Figure 10-63 Metro Mirror Session fully reversed, and completed

In this example, the secondary volumes are available for immediate production usage, and also replication back to the old master.

10.12 XIV asynchronous mirrors (Global Mirror)

To use Tivoli Productivity Center for Replication for any type of XIV Copy Services, including XIV's asynchronous mirror capabilities, you first need to create and define a new session, add the XIV volumes to that new session, and finally, activate the session.

The process for setting up Tivoli Productivity Center for Replication Global Mirror with XIV is nearly identical to what was already described for Tivoli Productivity Center for Replication Metro Mirror.

As noted previously, both the XIV pool and volumes must be defined using the XIV GUI or CLI before using Tivoli Productivity Center for Replication for this process. At the time of this writing XIV pools or volumes cannot be created from Tivoli Productivity Center for Replication.

10.12.1 Defining a session for asynchronous mirroring (Global Mirror)

Use the following steps to define a Tivoli Productivity Center for Replication session for asynchronous mirroring:

1. In the Tivoli Productivity Center for Replication GUI, navigate to the Sessions window and click **Create Session**. For details, see the similar process shown starting with Figure 10-35 on page 356.
2. When prompted for a session type, select **Asynchronous**; **Global Mirror**, as shown in Figure 10-64 on page 369. Click **Next** to start the process of creating this specific type of session.

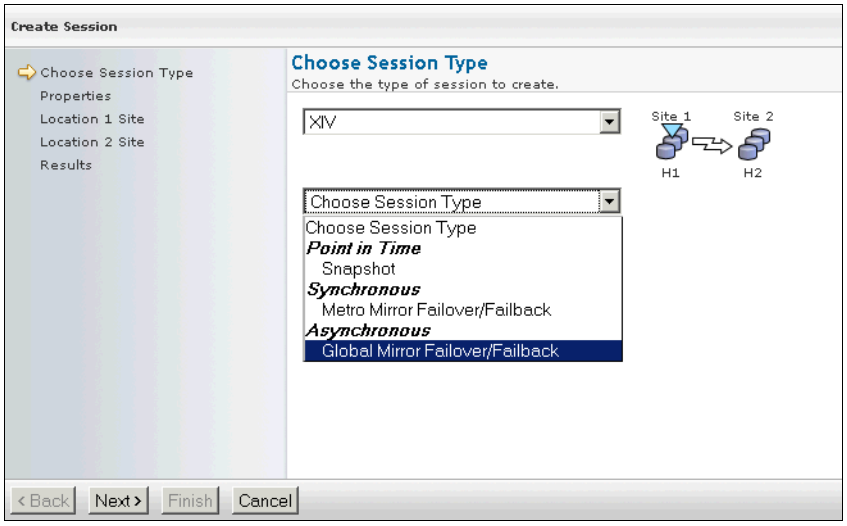


Figure 10-64 Tivoli Productivity Center for Replication Sessions panel; Asynchronous session type

3. Make the appropriate entries and selections in the panel shown in Figure 10-65.

   The difference between Metro Mirror and Global Mirror sessions is that for Global Mirror, Tivoli Productivity Center for Replication asks for the Recovery Point Objective (RPO) in seconds, and the selection box underneath asks for the scheduling interval.

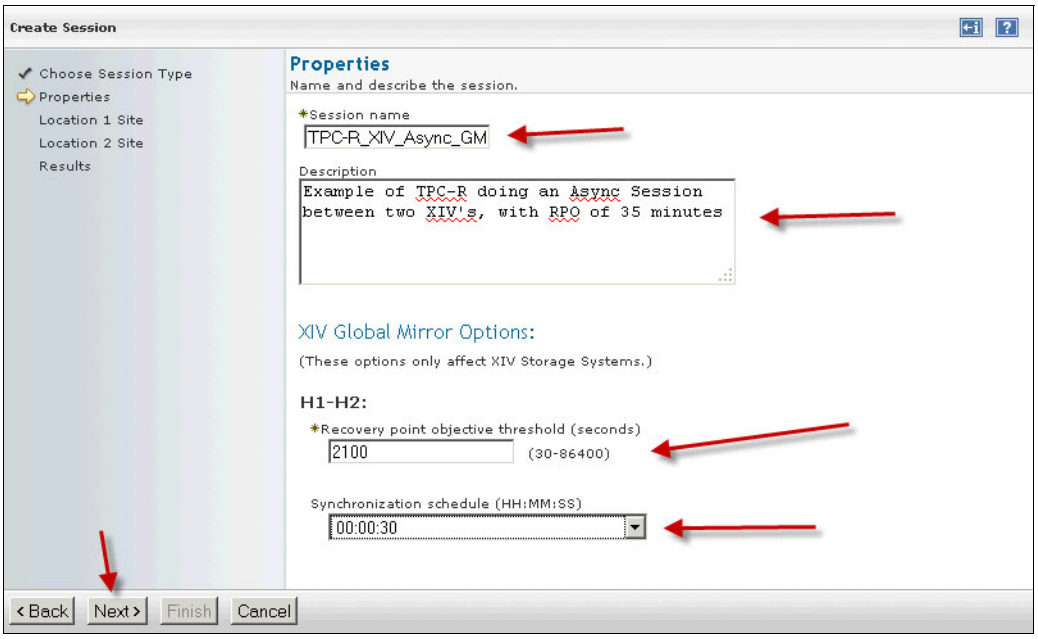


Figure 10-65 Session Wizard for asynchronous properties; RTO options

4. Click **Next** to proceed through the wizard's instructions to finish the process. This is the same process as Tivoli Productivity Center for Replication Metro Mirror, which is described in 10.11.1, "Defining a session for Metro Mirror" on page 354.

10.12.2 Defining and adding copy sets to a Global Mirror session

This second phase of the Tivoli Productivity Center for Replication process for Global Mirror, adding copy sets, is identical to what is described in 10.11.2, “Defining, adding copy sets to a Metro Mirror session” on page 358.

10.12.3 Activating the Global Mirror session

This is the third, and last, phase of the process. You are now ready to activate the Global Mirror session. From the Select Action list, select **Start H1->H2**, and click **Go** to activate the session. This is shown in Figure 10-66.

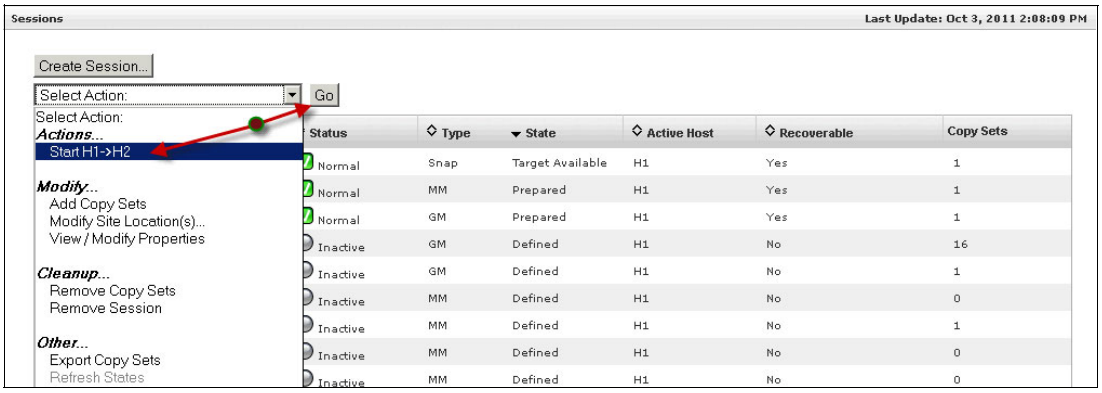


Figure 10-66 Tivoli Productivity Center for Replication session actions for Global Mirror

10.12.4 Suspending the Global Mirror session

Tivoli Productivity Center for Replication treats Global Mirrors in a similar fashion to Metro Mirrors. As described in 10.11.4, “Suspending the Metro Mirror (XIV Synchronous Mirror) session” on page 364, at some point you might want to suspend, if not reverse, the Global Mirror session. Follow the process in that section.

10.13 Using Tivoli Productivity Center for Replication to add XIV Volume Protection

Tivoli Productivity Center for Replication also has another protection function for storage volumes, whereby you can restrict certain volumes from other Tivoli Productivity Center for Replication Sessions or Copy Sets.

From the Storage Systems tab, select the XIV system and click **Volume Protection**, as shown in Figure 10-67 on page 371.

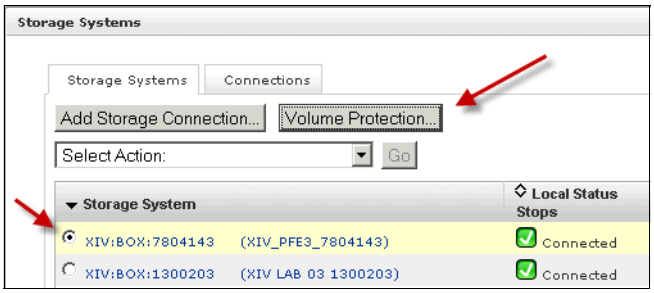


Figure 10-67 Volume Protection; select array and then click Volume Protection

Use the various pull-down lists shown in Figure 10-68 to successively select the pool and volumes. Using an asterisk (*) in the last input field returns a list of all the volumes in that pool. Optionally, you can use that field to filter the list of volumes returned.

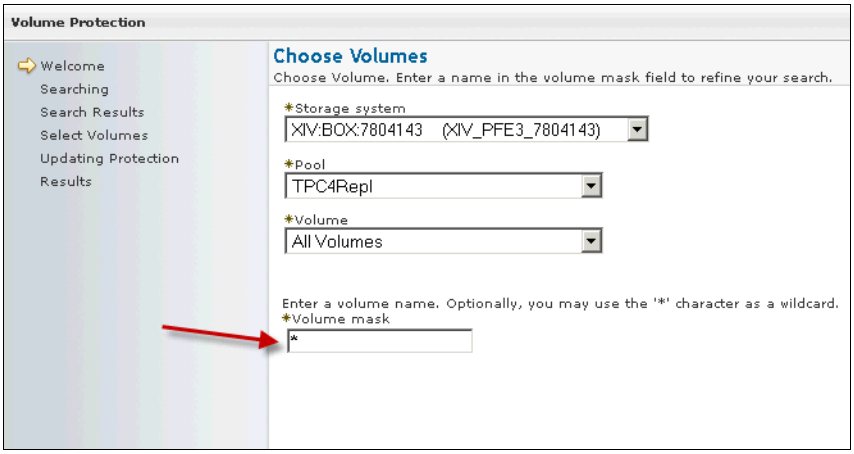


Figure 10-68 Tivoli Productivity Center for Replication volume selection for protection; wizard step 1

Click **Next** to display the volumes as shown in Figure 10-69, and select those you want to mirror.

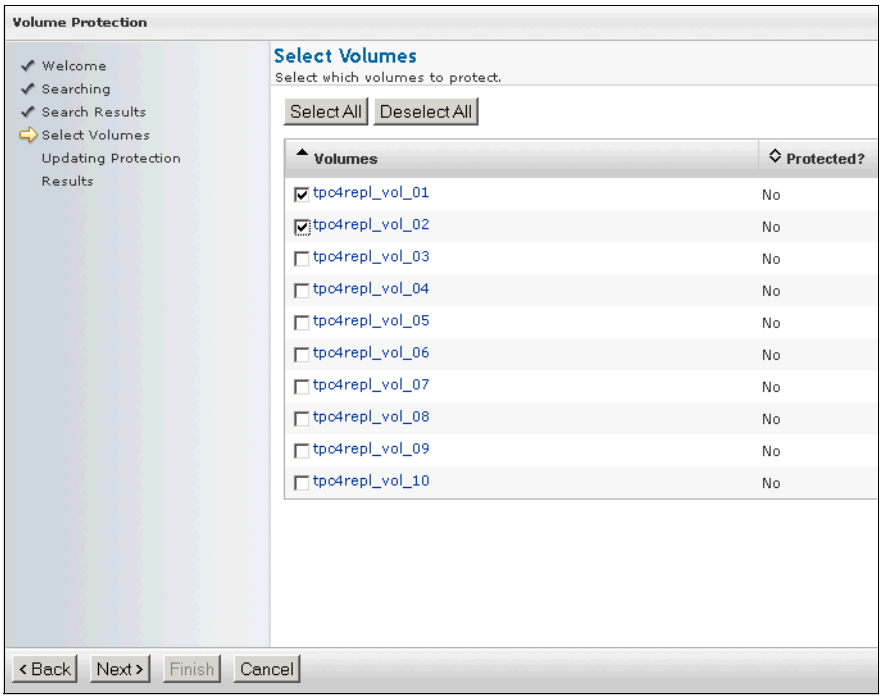


Figure 10-69 Tivoli Productivity Center for Replication volume selection for mirroring; wizard step 2

Click **Next**. Tivoli Productivity Center for Replication will now ensure that the selected volumes are protected from other Tivoli Productivity Center for Replication operations.

Important: Remember that these actions will only help inside the Tivoli Productivity Center for Replication system. Any administrator accessing the XIV GUI directly will not be informed of the volume protections. They will still see any snapshot or volume locks that are part of normal operations, but not any of the protections described here.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks publications

This document might be available in softcopy only:

- *DS8000 Copy Services for IBM i with VIOS*, REDP-4584
- *IBM Hyper-Scale in XIV Storage*, REDP-5053
- *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120
- *IBM Tivoli Storage Productivity Center V4.2 Release Guide,* SG24-7894
- *IBM XIV Storage System Architecture and Implementation*, SG24-7659
- *IBM XIV Storage System with the Virtual I/O Server and IBM i,* REDP-4598
- *RESTful API Support in IBM XIV*, REDP-5064
- *Using the IBM XIV Storage System in OpenStack Cloud Environments*, REDP-4971
- *XIV Storage System: Host Attachment and Interoperability*, SG24-7904
- *Solid-State Drive Caching in the IBM XIV Storage System*, REDP-4842
- *XIV Storage System in a VMware Environments*, REDP-4965
- *XIV Security With Data-At-Rest Encryption*, REDP-5047
- *XIV Gen3 with IBM System Storage SAN Volume Controller and Storwize V7000*, REDP-5063

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, drafts, and additional materials at the following website:

ibm.com/redbooks

Other publications

These publications are also relevant as further information sources:

- *IBM XIV Storage System Application Programming Interface,* GC27-3916
- *IBM XIV Storage System User Manual*, GC27-3914
- *IBM XIV Storage System: Product Overview*, GC27-3912
- *IBM XIV Storage System Planning Guide, SC27-5412-02*
- *IBM XIV Storage System XCLI Utility User Manual*, GC27-3915
- *IBM XIV Remote Support Proxy Installation and User's Guide*, GA32-0795

© Copyright IBM Corp. 2014. All rights reserved.

Online resources

These Web sites are also relevant as further information sources:

- IBM XIV Storage System Information Center and documentation:

  http://publib.boulder.ibm.com/infocenter/ibmxiv/r2/index.jsp

- XIV documentation and software:

  http://www.ibm.com/support/entry/portal/Downloads

- IBM XIV Storage System:

  http://www.ibm.com/systems/storage/disk/xiv/index.html

- IBM System Storage Interoperability Center (SSIC):

  http://www.ibm.com/systems/support/storage/config/ssic/index.jsp

- Implementing and managing Tivoli Productivity Center for Replication configurations:

  http://www.ibm.com/support/entry/portal/Documentation/Software/Tivoli/Tivoli_Storage_Productivity_Center_Standard_Edition

- Supported Storage Products Matrix for Tivoli Storage Productivity Center (v4.2.x):

  https://www.ibm.com/support/docview.wss?uid=swg27019305

- Tivoli Storage FlashCopy Manager:
  - http://www.ibm.com/software/tivoli/products/storage-flashcopy-mgr
  - http://publib.boulder.ibm.com/infocenter/tsminfo/v6

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

IBM XIV Storage System Copy Services and Migration

(0.5” spine)
0.475”<->0.873”
250 <-> 459 pages

IBM XIV Storage System Copy Services and Migration

Details of the copy services and migration functions

Offline initialization for sync and async mirroring

Generation 2 to Gen3 migration scenarios

This IBM Redbooks publication provides a practical understanding of the IBM XIV Storage System copy and migration functions. The XIV Storage System has a rich set of copy functions suited for various data protection scenarios, which enables clients to enhance their business continuance, data migration, and online backup solutions. These functions allow point-in-time copies, known as snapshots and full volume copies, and also include remote copy capabilities in either synchronous or asynchronous mode. These functions are included in the XIV software and all their features are available at no additional charge.

The various copy functions are reviewed in separate chapters, which include detailed information about usage, and also practical illustrations.

Finally, the book illustrates the use of IBM Tivoli Storage Productivity Center for Replication to manage XIV Copy Services.

This book is intended for anyone who needs a detailed and practical understanding of the XIV copy functions.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7759-04 ISBN 0738439266